University of Louisville
ThinkIR: The University of Louisville's Institutional Repository

Electronic Theses and Dissertations

5-2018

A framework for cardio-pulmonary resuscitation (CPR) scene retrieval from medical simulation videos based on object and activity detection.

Anju Panicker Madhusoodhanan Sathik
University of Louisville

Follow this and additional works at: https://ir.library.louisville.edu/etd

 Part of the Computer Engineering Commons

Recommended Citation

Madhusoodhanan Sathik, Anju Panicker, "A framework for cardio-pulmonary resuscitation (CPR) scene retrieval from medical simulation videos based on object and activity detection." (2018). *Electronic Theses and Dissertations.* Paper 2976.
https://doi.org/10.18297/etd/2976

This Doctoral Dissertation is brought to you for free and open access by ThinkIR: The University of Louisville's Institutional Repository. It has been accepted for inclusion in Electronic Theses and Dissertations by an authorized administrator of ThinkIR: The University of Louisville's Institutional Repository. This title appears here courtesy of the author, who has retained all other copyrights. For more information, please contact thinkir@louisville.edu.

A FRMEWORK FOR CARDIO-PULMONARY RESUSCITATION (CPR) SCENE RETRIEVAL FROM MEDICAL SIMULATION VIDEOS BASED ON OBJECT AND ACTIVITY DETECTION

By

Anju Panicker Madhusoodhanan Sathik
M.S., Computer Engineering and Computer Science,
University of Louisville, Louisville, KY

A Dissertation
Submitted to the Faculty of the
J.B. Speed School of Engineering of the University of Louisville
in Partial Fulfillment of the Requirements
for the Degree of

Doctor of Philosophy in Computer Science and Engineering

Department of Computer Engineering and Computer Science
University of Louisville
Louisville, Kentucky

May 2018

Copyright 2018 by Anju Panicker Madhusoodhanan Sathik

All rights reserved

A FRMEWORK FOR CARDIO-PULMONARY RESUSCITATION (CPR) SCENE RETRIEVAL FROM MEDICAL SIMULATION VIDEOS BASED ON OBJECT AND ACTIVITY DETECTION

By

Anju Panicker Madhusoodhanan Sathik
M.S., Computer Engineering and Computer Science,
University of Louisville, Louisville, KY

A Dissertation Approved On

April 20,2018

by the following Dissertation Committee:

Hichem Frigui, Ph.D., Dissertation Director

Xiang Zhang, Ph.D.

Olfa Nasraoui, Ph.D.

Juw Won Park, Ph.D.

Harry Zhang, Ph.D.

ACKNOWLEDGEMENTS

Thank God! I want to thank my family; I could have never completed this work without so much of sacrifice and support from them. Even from far away, they were always my greatest strength.

I would like to express my deepest gratitude to my advisor Dr. Hichem Frigui for his guidance and support throughout the duration of this research. I thank him particularly for his patience and motivation, for all the discussions, seminars and ideas which made the journey interesting. I could not possibly have had a better advisor than Dr. Frigui for my PhD.

Besides my advisor, I would like to thank the rest of my thesis committee Dr. Xiang Zhang, Dr. Olfa Nasraoui, Dr. Harry Zhang and Dr. Juw Won Park, for agreeing to serve in my committee.

I would also like to thank Dr. Adel Elmaghraby and Dr. Ahmed Desoky for their care and support, for giving me the opportunity to pursue my degree and for encouraging me to move on, when the going was tough. I want to express my regards to Dr. Aaron W. Calhoun for his help and guidance.

A special thanks to my friends and fellow members of Multimedia Research Lab for their collaboration, light talks, motivation and encouragement, which undoubtedly helped me keep the candle burning over the years.

ABSTRACT

A FRMEWORK FOR CARDIO-PULMONARY RESUSCITATION (CPR) SCENE RETRIEVAL FROM MEDICAL SIMULATION VIDEOS BASED ON OBJECT AND ACTIVITY DETECTION

Anju Panicker Madhusoodhanan Sathik

April 20, 2018

With the increasing wealth of videos available today, automatic classification and retrieval of videos is gaining extreme importance. A vast amount of research has been done in the field of content based video retrieval, in the past decade, where video representations were mainly drawn from three modalities text, audio and visual - and their combinations with several classification techniques. An important branch of content based video retrieval is activity detection from videos.

In this thesis, we propose a framework to detect and retrieve CPR activity scenes from medical simulation videos. Medical simulation is a modern training method for medical students, where an emergency patient condition is simulated on human-like mannequins and the students act upon. These simulation sessions are recorded by the physician, for later debriefing. With the increasing number of simulation videos, automatic detection and retrieval of specific scenes became necessary. Our application is specific to detecting Cardio Pulmonary Resuscitation (CPR) activity and the breathing bag activity, from the medical simulation videos. The proposed framework for scene retrieval, would eliminate the conventional approach of using shot detection and frame segmentation techniques.

Firstly, our work explores the application of Histogram of Oriented Gradients in three

dimensions (HOG3D) to retrieve the scenes containing CPR activity. Human action recognitions are by far mostly studied under well controlled laboratory environments, without background disturbances, camera movements or occlusions. The proposed approach would be able to detect specific activity even when there are multiple people performing different actions, camera movements and several background disturbances.

Secondly, we investigate the use of Local Binary Patterns in Three Orthogonal Planes (LBP-TOP), which is the three dimensional extension of the popular Local Binary Patterns, for activity based scene retrieval from videos. This technique would also be used in the proposed framework as a robust feature that can detect specific activities from scenes containing multiple actors and activities.

Thirdly, we propose an improvement to the above mentioned methods by a combination of HOG3D and LBP-TOP. Since features from different modalities can provide complementary information, we use decision level fusion techniques to combine the features. We prove experimentally that the proposed techniques and their combination out-perform the existing system for CPR scene retrieval.

Finally, we devise a method to detect and retrieve the scenes containing the breathing bag activity, from the medical simulation videos.

We use a Support Vector Machine (SVM) classifier for classification of the video volumes into CPR activity scenes or non-CPR activity scenes. We also experiment with K-Nearest Neighbours (KNN) and Artificial Neural network (ANN) classifiers and devise decision level fusion techniques for CPR activity detection. Finally, we have developed a graphical user interface (GUI) that gives a statistical display of the correctness of the CPR activity performed and enables the user to directly view the desired scenes.

The proposed framework is tested and validated using eight medical simulation videos and the results are presented.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

The popularity of surveillance cameras and personal video recorders contributed heavily to the enormous amount of video data these days. According to recent statistics, over 100 hours of video are uploaded to YouTube alone every minute [8]. The superfluity of videos available both on the internet and elsewhere has made manual annotation of the videos almost impractical. Consequently, widespread research has been going on to automatically categorize, analyze, classify, index, and search the large collections of videos. This area of research is commonly referred to as content based video (CBV) analysis [9–14]. A typical CBV analysis system consists of the following three main steps:

- Pre-processing - Videos may require different pre-processing techniques depending on the quality of video recording and the application at hand. Background subtraction, contrast normalization, filtering, sampling etc. are different types of pre-processing methods generally used for video analysis [15–17].

- Feature Extraction - It is important to extract the appropriate features to represent the content of a video. Video features are typically extracted from textual [18, 19], audio or visual modalities [20–22]. In general, a combination of multiple features from different modalities will be needed to represent the complex content of a video [23]. In such cases, fusion of different features from different modalities are used.

- Feature Learning - Once the features are extracted, the desired system (classifier, search engine, indexing application etc.) is built by analyzing and learning patterns within the features. Existing learning algorithms use supervised [24, 25], semi-supervised [26, 27], or unsupervised [28, 29] techniques for building the system.

One of the most common tasks in CBV analysis is the classification of video scenes based on the performed actions [25, 30]. Then, at a higher level, videos can be categorized based on the frequency of occurrence of particular actions. For example, in the case of online movie classification, say we want to classify movies into horror movies or action movies. This can be done by finding the frequency of occurrence of horror scenes followed by screams, or fight scenes respectively, and then classifying the movies accordingly. Action recognition is therefore crucial for organizing and retrieving online videos [31, 32].

Another important application of activity detection is in video surveillance [33, 34]. According to IHS (Information handling Services Markit) in 2014, there were 245 million professionally installed video surveillance cameras active and operational globally [35]. The main purpose of installing surveillance cameras was to cut down crime. However, these video recordings are very long, and manual detection and retrieval of scenes containing specific activities is tedious and mundane. Thus, automatic real-time detection of certain activities such as a fight scene, robbery, or suspicious activity [36–38] in police surveillance videos, can be very helpful.

Activity detection in the closed-circuit television surveillance of elderly people is another application that can benefit from CBV analysis [39]. Specific tasks in this domain include fall detection [40,41], monitoring home medical device operation etc. Similarly, activity detection from videos can be used for educational and medical purposes like studying human behavior. Researchers, analysts, historians and journalists also need efficient and accurate retrieval of archived video data for several applications.

Automatic retrieval of scenes containing particular activities find tremendous applications in sports and entertainment industry as well [42–45]. Sports highlights, replays, classification, all require automatic action detection and retrieval techniques. CBV analysis finds increasing use in gaming and animation industry that rely on synthesising realistic humans and human motion. Gaming industry tend to produce a large variety of motions [46] to facilitate augmented reality and human machine interactions, whereas movie industry focuses on producing high-quality animations [47, 48].

Human Activity Recognition (HAR) is an important area within CBV research [49]. HAR is difficult, owing to several reasons such as the high dimensionality of the features representing the video data, large intra-class variability due to difference in scale, illumination changes, camera movements and also the resolution and quality of the video recording.

HAR in medical videos has been gaining attention in the past few years. It finds tremendous use in patient monitoring videos [40,50,51], fall detection in elderly monitoring videos [41], early detection of many diseases like hand tremor or neonatal epilepsy [50,52], and also in medical simulation videos which are recorded for educational purposes.

Studies published in the *Journal of the American Medical Association* in January 2005 [53] show that CPR can be incorrectly performed even by trained practitioners. Medical simulation is a recent advanced methodology for training medical students to improve patient safety, strengthen interdisciplinary and clinician-patient interactions. It allows the acquisition of clinical skills through deliberate practice rather than simply acquiring theoretical knowledge.

In this work, we focus on CPR activity detection in medical simulation videos. This work is in collaboration with the Simulation for Pediatric Assessment, Resuscitation, and Communication (SPARC) group, within the Department of Pediatrics at the University of Louisville. SPARC was developed to teach pediatrics faculty, fellows, and residents how to respond to medical crises. The objective of SPARC is to enhance the care of infants and children by using simulation-based educational methodologies to improve patient safety, strengthen interdisciplinary and clinician-patient interactions.

The SPARC simulation sessions involve 4 to 9 people and last approximately 15 minutes to one hour. Human-like mannequins that have respiration and heartbeat and respond to treatment with virtual drugs, are used for these simulations. After each such session, the physician would manually review and annotate the recording, and then debrief the trainees on the session. Video assisted debriefing allows participants to reflect on their experience, teaching them to be more efficient and productive during such real life scenarios. With the increasing number of simulation sessions, the physicians find that (1) the

manual process of review and annotation is labor intensive; (2) retrieval of specific video segments is not trivial; and (3) there is wealth of information waiting to be mined form these recordings. Providing the physician with automated tools to segment, semantically index and retrieve specific scenes from a large database of training sessions will enable him/her to (1) immediately review important sections of the training with the team, (2) allow more efficient debriefing session with the team of trainees, and (3) identify similar circumstances in previously recorded sessions.

CPR is a combination of techniques designed to pump the heart to keep the blood circulating and deliver oxygen to the brain, in case the heart stopped pumping (cardiac arrest) until definitive measures can stimulate the normal working of the heart. The Breathing Bag or Bag Valve Mask (BVM) is a required component of the CPR kit for trained professionals and is often found as standard equipment in emergency rooms and other critical care settings. The BVM consists of a flexible air chamber attached to a face mask through a shutter valve. The flexible air chamber is squeezed to force feed air or oxygen into the lungs of the patient, in order to inflate them under pressure, thus manually providing a positive pressure ventilation. When the bag is released, it self inflates from the other end drawing in either ambient air or a low pressure oxygen flow supplied by a regulated cylinder, while also allowing the patient's lungs to deflate. Bag-valve masks may be of varying sizes so as to fit infants, children, or adults. The face mask size may be independent of the bag size; for example, a single pediatric-sized bag might be used with different masks for multiple face sizes, or a pediatric mask might be used with an adult bag for patients with small faces. A picture of breathing bag used in our experiments is shown in figure 1.1.

Figure 1.1: Image of a breathing bag

In an emergency room simulation, a typical CPR procedure for an adult consists

of either 15 chest compressions followed by 2 breaths or 30 chest compressions followed by 2 breaths depending on the number of people present. The ratio only holds if there is no endotracheal tube. The CPR and breaths are given in a cyclic manner. If the patient is intubated then the ratio must be 100 compressions per minute and 20 breaths per minute given asynchronously. If there is cardiac dysrhythmia (such as *ventricular tachycardia* and *ventricular fibrillation*) present, an external defibrillator will be used after a specific number of CPR cycles.

In this dissertation, we developed a framework to automatically detect and retrieve scenes containing CPR activity from medical simulation videos. We also devised a method to detect the breathing bag action that follows the CPR activity. By measuring the ratio of the number of CPR compressions to the breathing bag compressions, we report an estimate of the correctness of the procedure performed by different trainees in the simulation session.

The main contributions of this dissertation can be outlined as follows:

1. We developed an adaptation of spatio-temporal Histogram Of Gradient orientation features (HOG3D) [3]. HOG3D uses the orientation of gradients to capture the structure of the motion in three dimensions.

2. We developed a method to use spatio-temporal texture features to capture the dynamic texture that appears during the up-down rhythmic movement of a CPR action cycle. We use Local Binary Patterns on Three Orthogonal Planes (LBP-TOP) [54] for this purpose.

3. We implemented a decision-level fusion of HOG3D and LBP-TOP to obtain higher accuracy as compared to either of the methods individually. We train and test with three different classifier models and report the results.

4. We devised a method to detect the breathing bag activity by detecting the change in area of the bag, when the breathing bag activity is performed.

5. We developed a GUI prototype to enable the user to retrieve the CPR activity scenes and also display the correctness of the CPR activity.

An important contribution of this work is how we achieve the above mentioned tasks without using the conventional modules like key-frame extraction, shot segmentation, and scene segmentation. Another key aspect of this work is that we present experimental results on a larger dataset compared to previous studies. This research aims at reducing the false rejection, false acceptance, and, training and testing time for reliable recognition and retrieval of scenes specific to CPR activity.

The remainder of this dissertation is organized as follows. In chapter 2, we provide an overview of related work on activity recognition and detection techniques, and existing work on CPR scene retrieval system. In chapter 3, we propose the CPR scene identification system using spatio-temporal features. In chapter 4, we present the experimental results and analysis, and describe our graphical user interface (GUI). Finally, in chapter 5, we conclude and discuss about potential future work.

CHAPTER 2

BACKGROUND AND LITERATURE REVIEW

In this chapter, we present an overview of the state-of-the-art methods for activity detection and recognition in general, as well as in the medical domain. We also present an overview of prominent decision-level fusion techniques to combine the outputs of multiple algorithms.

In general, methods for modeling and recognizing actions can be classified into non-parametric, volumetric, or parametric time series approaches. In the non-parametric approach, first, low-level visual features are extracted for each video frame. Then, these features are matched to pre-computed templates [55–58] that correspond to different actions. Templates can be either two dimensional (2-D) or three dimensional (3-D). The 2-D template modeling generally consists of a blob identification step using background subtraction or tracking, followed by computation of flow-based features for each identified spatial location. The flow features within each segment are averaged into a single frame. The average-flow frames, within an activity cycle, form the template for each action class [5, 57]. The main limitation of the template-based approach is that they tend to lose their discriminative power for complex activities due to averaging. To model 3-D objects, the 2-D object blobs, or contours in the (x, y) spatial space, are stacked together to encode shape and motion characteristics in the 3-D (x, y, t) space [59]. Manifold learning algorithms, which are methods for reducing the high dimensionality of video data for action recognition tasks [60, 61], also fall under this category.

In the volumetric approach, features are extracted over a 3-D volume of pixel intensities [1, 3, 62]. The volumes can be extracted based on spatio-temporal filtering [1], part-based detection [4, 63], or sub-volume matching [3]. In spatio-temporal filtering, the

video volume is filtered using a large filter bank at various orientations and spatial scales. The filter's responses along eight different orientations for a hand-waving action is illustrated in figure 2.1. Once the motion salient regions are identified, actions are either recognized using mean-shift tracking of the salient regions, or by creating a histogram of energies (HOE) as feature representing the action.

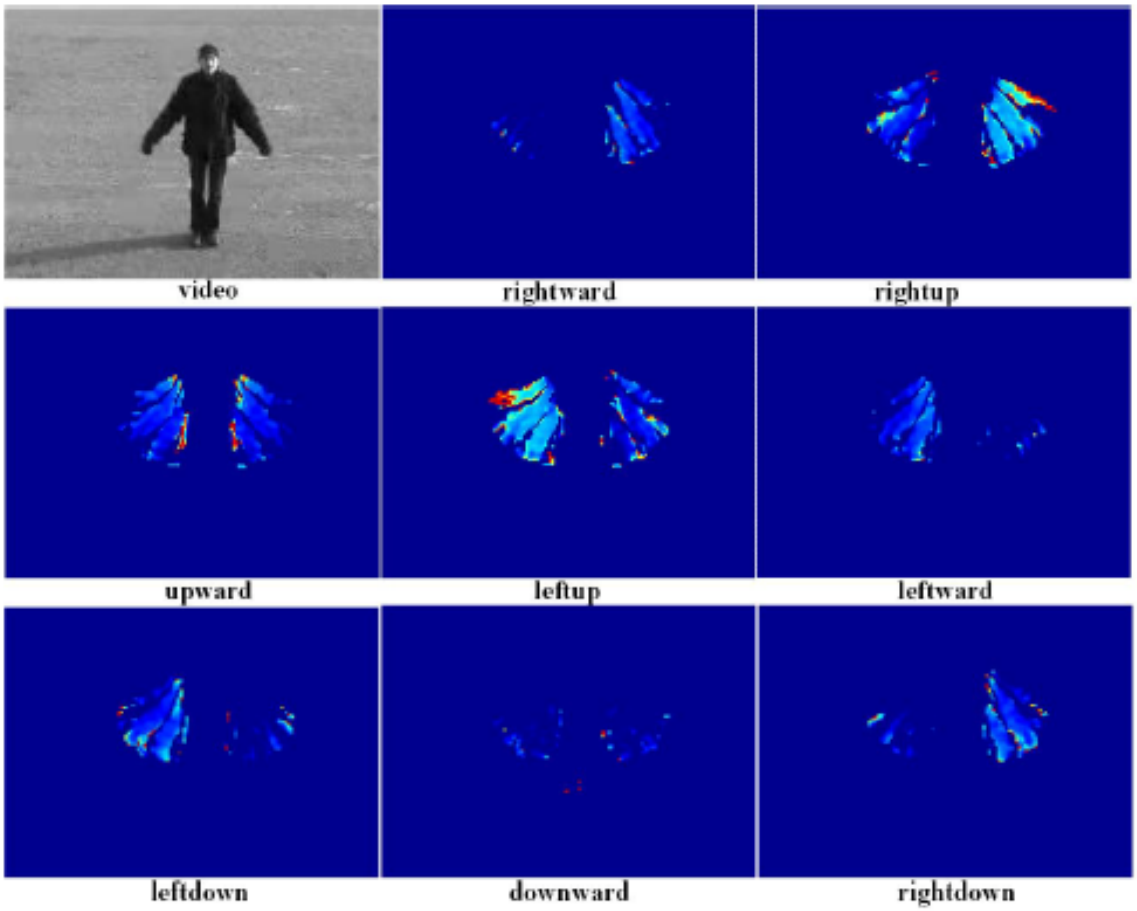


Figure 2.1: Low-level features - Motion saliency features of eight orientations (courtesy [1])

In part-based detection, a video volume is considered as a collection of local parts and each part consists of a distinctive motion pattern which contributes to the activity [4]. In [2], Laptev and Lindeberg proposed a 3-D generalization of scale-space representations, where spatio-temporal gradients are computed for each level of scale-space representation of the video. The gradients within a neighborhood are combined to yield stable estimates of a spatio-temporal second-moment matrix. Local features are then derived from these

smoothed estimates of gradient moment matrices.

The parametric approach for activity detection and recognition attempts to model the temporal dynamics of motion. The specific parameters for a class of action are estimated from the training data [64–67]. The two most popular parametric approaches are the Hidden Markov Model (HMM) and the Linear Dynamical Systems (LDS). In HMM, the state space is considered to be a finite set of discrete points and the temporal evolution is modeled as a sequence of probabilistic transitions from one discrete state to another [7, 64]. HMMs are efficient for modeling time-sequence data and are useful for both their generative and discriminative capabilities. They are also well suited for tasks that require recursive probabilistic estimates. Linear dynamical systems are a more general form of HMMs where the state space is not constrained to a finite set of symbols but can take on continuous values [66, 67]. LDSs are commonly used for learning patterns from high-dimensional time series data.

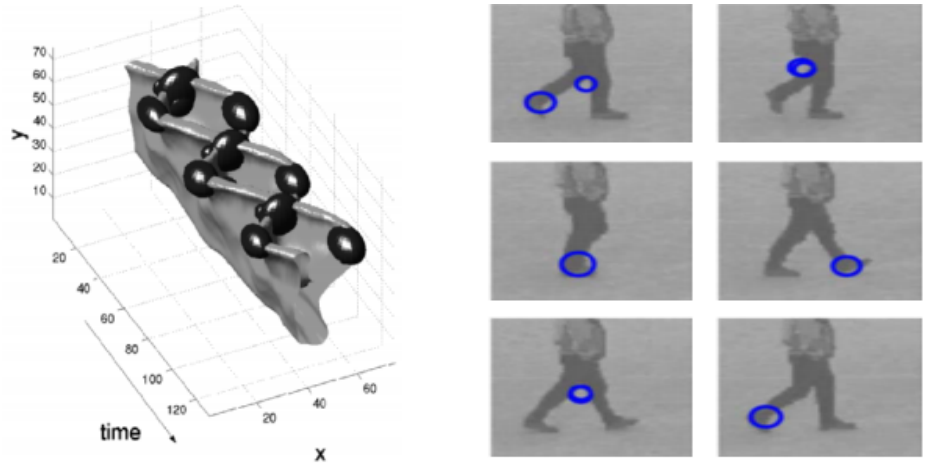
In this chapter, we start by reviewing the state-of-the-art methods for detecting general human activities. This task consists of the detection of day to day actions like running, jogging, walking etc.Typically, these methods are developed and tested on datasets where videos are recorded under controlled environments such as a single actor is performing the action without significant background changes or camera movement. Next, we focus on specific activities within the medical domain. Activities within this category follow a particular procedure or pattern and require a deeper understanding of the medical procedure [51]. These will generally have multiple actors and multiple activities being performed simultaneously. An example of this application is the detection and retrieval of (CPR) scenes from medical simulation videos.

2.1 General Human Activity Detection Methods

Depending on the type of features used and the method of detection, general human activity detection can be categorized into three main groups: 1) detection using low level features; 2) detection using high-level features; and 3) tracking based detection.

2.1.1 Activity detection using low-level features

Typically, low-level features are local descriptors or interest points extracted at every frame or from 3-dimensional space-time volumes. Examples of such features include Harris3D [68], Cuboids [69] and Spatio Temporal Interest Points (STIP) [2] to locate spatio-temporal interest points. For example, the STIP detectors (figure 2.2) are computed based on the detection of spatio-temporal corners. These points of interest are determined as those points that exhibit a high variation of image intensity in all three directions (x, y, t). Figure 2.2a(a) shows the 3-D plot of the detected interest points, illustrated by ellipsoids on a level surface of a leg pattern (shown upside down) and figure 2.2b(b) shows the actual interest points overlayed on the frames from the original walking sequence.



(a) (b)

Figure 2.2: Detected spatio temporal interest points for the leg movement in walking action: (a) 3-D plot of interest points (ellipsoids) on a level surface of a leg pattern shown upside down (b) STIP points overlayed on actual frames from a video of walking sequence (courtesy [2])

After detecting the interest points, a typical procedure is to extract a cuboidal volume of interest (VOI) around the interest point, and find spatio temporal descriptors, such as HOG/HOF [70], HOG3D [3, 4] etc. within the VOI. The extracted descriptors are quantified with a pre-learned code book, and input videos are typically modeled with Bag

of Visual Words [71]. These local descriptors are somewhat successful in capturing and representing local and repeatable properties of actions within a video. These properties make local descriptors robust to intra-class variability and deformation to a certain degree [63]. However, these local descriptors cannot discriminate between activities with different high-level motions effectively.

Histogram of Oriented Gradients (HOG) is a 2-D descriptor that has been used extensively for people detection, object detection, and activity recognition. Dalal and Triggs [72] explain how a dense histogram of gradient orientations can capture edge information which aids in efficient detection of pedestrians. HOG can also be used in combination with other temporal features for activity recognition. For instance, Laptev et al. in [70] introduced HOG/HOF descriptors which is a fusion of the 2-D HOG and histograms of optical flow (HOF). The two histograms are concatenated to form one descriptor. Both descriptors are computed in the space-time neighborhood of the detected interest points. Another descriptor that proved to be effective is the Histogram of Oriented Optical Flow (HOOF) features. HOOF was proposed in [73] to represent human activities using nonlinear dynamic systems of HOOF time series. The HOOF features essentially capture the distribution of optical flow. It is independent of the scale or direction of motion.

HOG and several of its variants [74, 75] were successfully used for local feature representations as well as for dense description of objects and images. In [3], Klaser *et al.* proposed the 3-D HOG features (referred to as HOG3D), which are fast and efficient to compute and are able to combine motion and appearance into one representation. This is a 3-D extension of the HOG features into the temporal dimension and is robust to changes in illumination. Figure 2.3 gives an overview of the HOG3D descriptor computation. For each video volume, a sparse set of spatio-temporal interest points are obtained using Harris3D ([68]) operator. The support region around the interest points are determined and these regions are divided into grids. Figure 2.3(a), shows the spatio-temporal region of size (h_s, w_s, l_s), around an interest point. This region is divided into a $M \times M \times N$ grid, (here $M = N = 2$). Each sub-volume C_i is divided into $S \times S \times S$ sub-blocks b_j. The

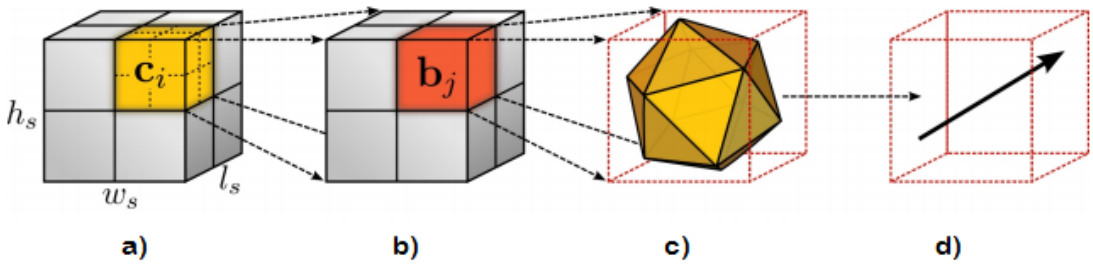
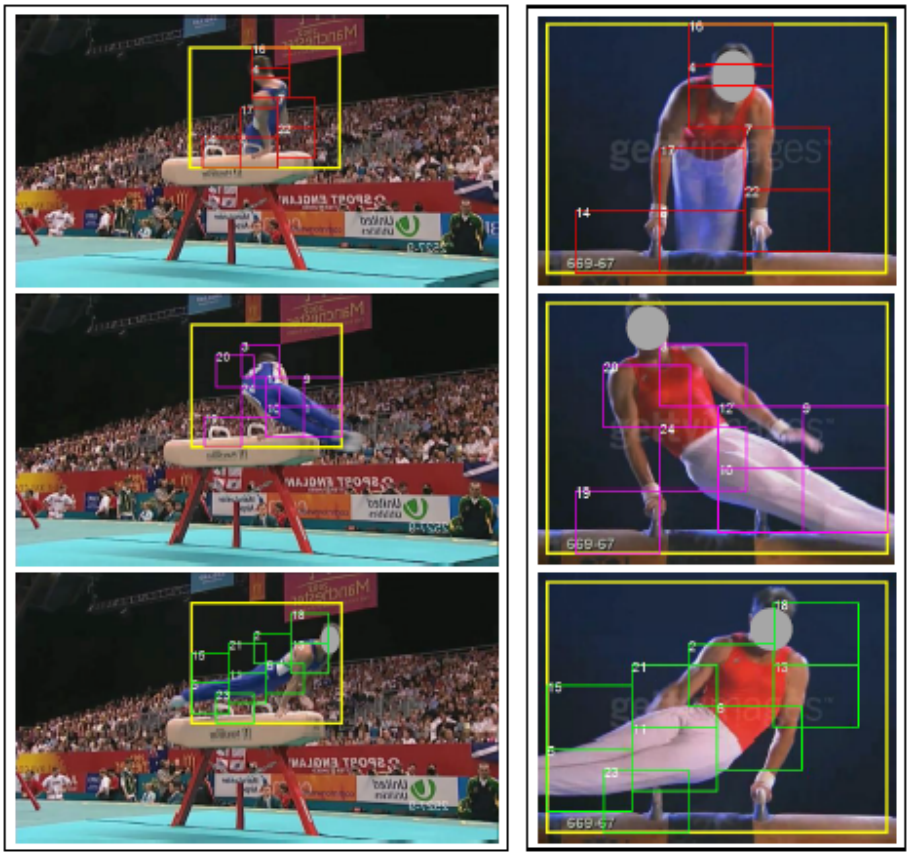


Figure 2.3: Overview of the HOG3D descriptor computation: (a) Support region around a point of interest divided into grid, gradient orientation histogram computed at each sub-volume in the grid is concatenated to form final histogram; (b) each histogram is computed over a grid of mean gradients; (c) each gradient orientation is quantized using regular polyhedrons; (d) each mean gradient (courtesy [3])

mean gradient is computed at each of the sub-blocks b_j in the grid (figure 2.3(d)). This gradient is projected through a 20-D polyhedron to form 20-D histogram (figure 2.3(c)). The histograms from all sub-blocks are concatenated to form the final feature vector. For activity detection using HOG3D, features are sampled at multiple spatial and temporal scales. Each video sequence is then represented using a *bag-of-words* representation of the sparse space-time features. Finally, a non-linear SVM classifier with χ^2- kernel is used for classification.

In [4], a spatio temporal deformable part model (SDPM) was proposed. The SDPM captures intra-class variations as a deformable configuration of parts. This model consists of a root filter and several part models defined by the corresponding part filters. The root filter captures the overall information of the action cycle while the part filters capture the spatio temporal configuration of body parts. SDPM uses HOG3D as features for the action cycle volume. Figure 2.4 shows an example of "Swing Bench" action which is modeled using several parts across three temporal stages. The large yellow rectangle indicates the area under the root filter and smaller rectangles represent the parts. For each action model, the most discriminative 3-D sub-volumes are automatically selected as parts, and the spatio-temporal relationship between their locations are learned. This model focuses on the most discriminative parts of each action, and therefore adapts to intra-class variation and shows robustness to clutter.



(a) training (b) testing

Figure 2.4: Action detection using spatio temporal deformable part model: The yellow rectangle represents root filter and the smaller rectangles represent part filters. Detection across three temporal stages is shown (courtesy [4])

In [54], the authors proposed an extension of the local binary patterns (LBP [76]), called volume local binary patterns (VLBP), to recognize facial expressions as dynamic textures. LBP is computed in a 3×3 cell by comparing the gray pixel intensities of every neighboring pixel to the center pixel. If the gray value is larger, then the corresponding position is assigned a value 1, else 0. The computation of VLBP is similar to that of local binary patterns. In particular, the intensity of the center voxel is compared with the intensities of the neighboring voxels (from a space-time volume rather than a single frame). The VLBP feature vector tends to be very high dimensional depending on the number of neighborhood points P, which is typically 8 or 4. For instance, for an 8-point

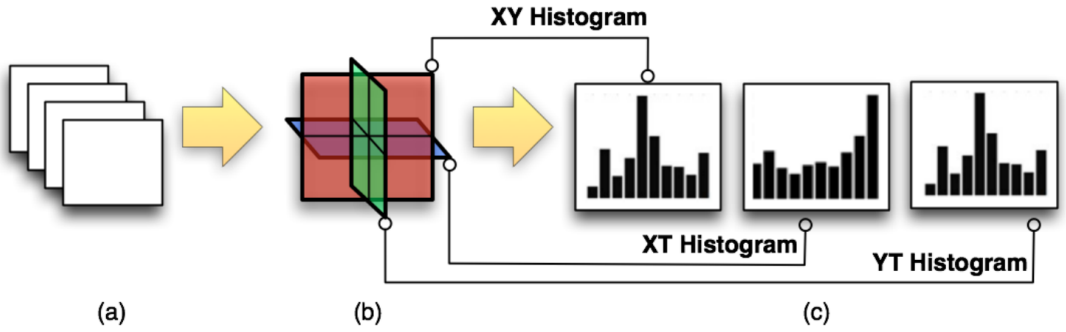


Figure 2.5: Computation of LBP on a 3-D volume as a concatenation of three LBP's extracted from Three Orthogonal Planes

neighborhood, the VLBP will have a 24 bit code word length, which leads to a total of 2^{24} different patterns. It has been experimentally proven that this large feature pool can be significantly reduced to a smaller subset for the task of action recognition [77, 78]. The feature vector can be made shorter by reducing the value of P, but with loss of information. Also if the chosen time interval $L > 1$ for VLBP computation, all frames with time variance less than L will be discarded. The VLBP descriptor can be simplified by concatenating the local binary patterns on three orthogonal 2-D planes: XY, XT and YT planes, while considering only the co-occurrence statistics in these directions as shown in Figure 2.5. This feature is referred to as LBP-TOP.

In [55], the VLBP and optical flow descriptors were combined to produce an efficient descriptor, called motion binary patterns (MBP). The MBP encodes the intensity variations in the neighborhood of each pixel as binary representations. A feature descriptor is created by computing the histogram of all the binary patterns obtained for the given video.

As outlined earlier, several formulations of spatio temporal descriptors have been applied to video representation. These features are either derived from spatial orientation of intensities, motion information, or 3-D textural analysis. However, the relevance of these different types of features and their mutual combinations are still under utilized. In several works, it has been proven that the integration of information from different modalities

provides a substantial improvement of the classification accuracy [79, 80].

2.1.2 Activity detection using high-level features

Activity detection methods based on high level features are made up of global templates of actions. Several high level features and models use human shape masks and silhouette information to represent the dynamics of the human body. For instance, in [81], Bobick and Davis use shape masks from difference images for detecting human actions (Figure 2.6) using two *temporal template* representations: *motion-energy image* (MEI), which is a binary value that represents where motion has occurred in an image sequence, and a *motion-history image* (MHI), which weight these regions according to the point of time they occurred (the more recent occurrences get higher weight). MEI is computed from the cumulative binary motion images, generated using image differencing, from the start frame to the end frame of the action sequence. In MHI, the pixel intensity is represented as a function of the temporal history at that point. This gives a scalar-valued image where more recently moving pixels appear brighter. Thus MEI and MHI behave as two components of a vector image that can encode motion properties in a spatially indexed manner. The idea of temporal templates was first introduced in [81]. Figure 2.6 shows sample key frames from an aerobics video, and the respective motion energy and motion history images.

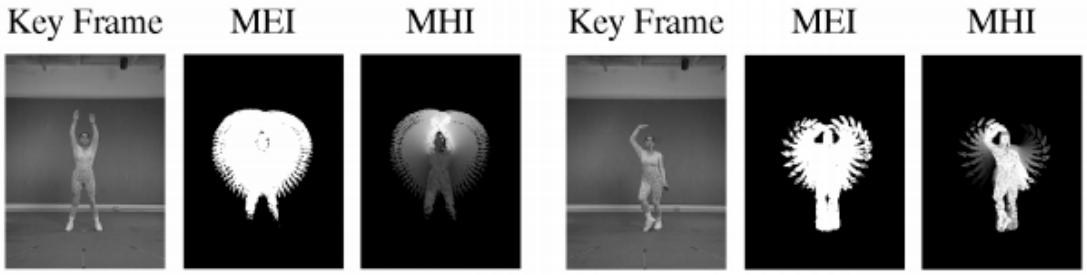


Figure 2.6: Shape masks from different images for computing motion history image (MHI) and motion energy images (MEI) (courtesy [5])

In [6], an action model based on space-time shapes from silhouette was introduced. Human actions in video sequences are regarded as 3-D shapes induced by the silhouettes in

the space-time volume. The silhouette is computed using background subtraction (Figure 2.7). The properties of the solution to Poisson equation is used here to extract features like local saliency, action dynamics, shape and structure. A high dimensional feature vector is used to represent chunks of 10 frames. A sliding window in the temporal dimension is used to compute features for detection, during classification.

Figure 2.7: Space-time volumes for action recognition based on silhouette information (courtesy [6])

In [82], the authors proposed the Action Bank technique that applies a large set of action detectors to the input video. The responses of these detectors are then used as a semantically rich representation. Action bank stores a large set of individual action detectors at various scales and view points. The action bank representation is a concatenation of volumetric max-pooled detection volume features from each detector. This feature representation, when paired with simple SVM classifiers, is shown to produce good results.

2.1.3 Activity detection using Tracking

Trajectories of body movements are common features used to identify the performed actions. The objective of tracking is to establish correspondence of action between consecutive action cycles based on features related to position, velocity, shape, texture, and color. In particular, human body or parts of the body are identified, segmented, and tracked to recognize the activity. Tracking can be done using a Kalman filter or a particle filter [83,84].

Typically, low-level features are used to derive the trajectory descriptors to represent the local motion in the video [85]. Once the trajectory features are obtained, a classifier (eg. SVM or HMM) can be trained and used for detection.

Since trajectory based activity detection methods are based on local features, they do not consider the global constraints in the space-time volume. A method to construct a neighborhood topology, both in the spatial and temporal domains, using a supervised manifold learning algorithm was proposed in [86]. This approach solves the generalized eigenvalue problem to obtain the best projections that not only separate data points from different classes, but also preserve local structures and temporal pose correspondence of sequences from the same class.

A framework for tracking long sequence of activities in videos using a parametric approach was described in [87], where activities are represented mathematically using dynamical models defined on the shape of the contour of the human body. This method is capable of tracking long multi-activity sequences including time instances of change from one activity to the next.

Another method to detect actions from dense trajectories obtained by sampling dense points from each frame and tracking them based on displacement information from a dense optic flow field, is described in [88]. In this approach, trajectories in the video are clustered and an affine transformation matrix is computed for each cluster. In addition to displacement vectors, the final trajectory descriptor contains elements of the affine transformation matrix for its assigned cluster center.

2.2 Activity Detection for Medical Applications

Cardio Pulmonary Resuscitation (or CPR) activity detection finds increasing use in medical activity detection from videos. In [7], the authors proposed an architecture for retrieving scenes that contain CPR activity from medical simulation videos. This method is shown in Figure 2.8.

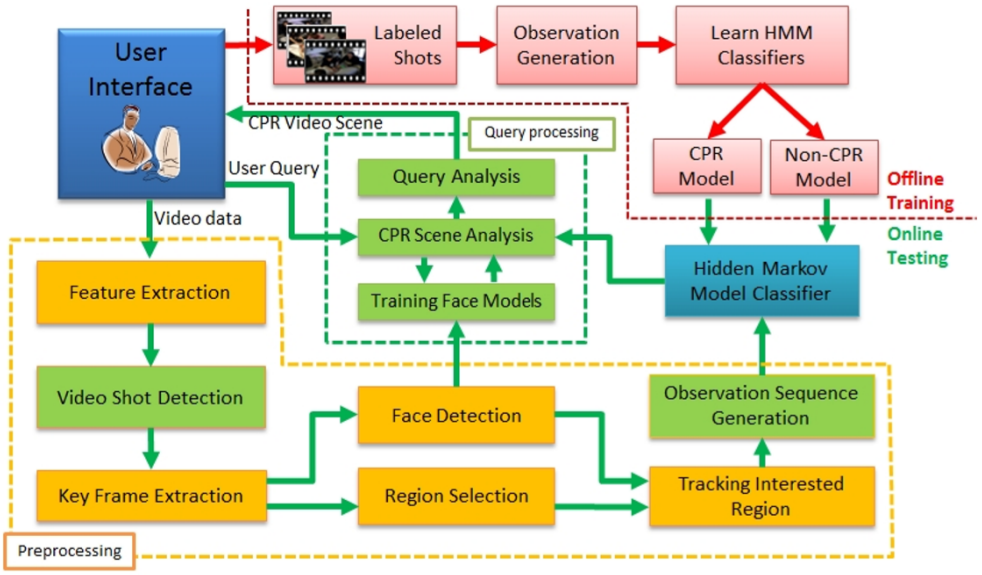


Figure 2.8: Architecture of the existing CPR scene retrieval system (courtesy [7])

The video is initially subjected to shot boundary detection by evaluating the similarity of visual features between adjacent frames. A combination of color, edge and motion features are used to evaluate the similarity measure for shot boundary detection [89, 90]. After video shot detection, a representative frame or key frame is chosen at a predefined temporal location in the shot to represent the salient information from the shot. The selected key frame is then segmented using the *NCut* algorithm [91]. Each of the segments is then passed through a skin detector. From the identified skin regions, the region of interest (hands of the person performing CPR) is selected. The average optical flow of the identified region is calculated as the motion features as shown in figures 2.9 and 2.10. The motion features are computed for each region of interest within each consecutive frame within a given shot. To discriminate between skin regions that are involved in CPR and other skin regions, a Hidden Markov Model (HMM) classifier [92] is developed and trained using the motion features.

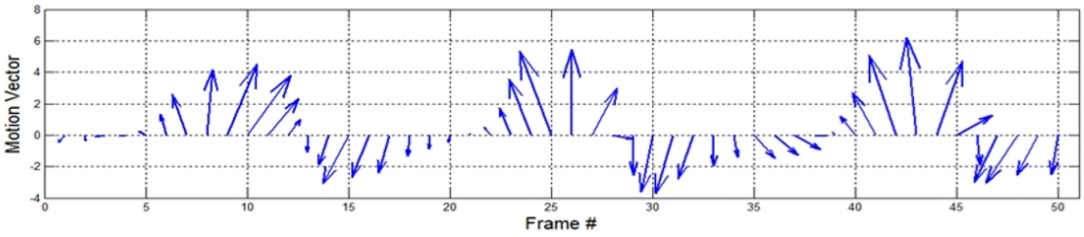


Figure 2.9: Average Optical flow of a CPR sequence (courtesy [7])

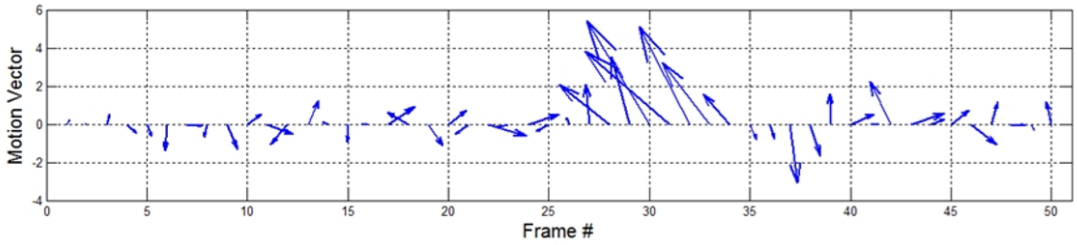


Figure 2.10: Average Optical flow of a non-CPR sequence (courtesy [7])

The HMM classifier produces a sequence of random observation vectors at discrete time according to an underlying Markov chain. A discrete HMM classifier with two models, one for CPR sequences and another one for non-CPR sequences is used in this implementation.

In [93] Semeraro *et al.* devise a means to improve the quality of the CPR by using a mini-VERM (Virtual Reality Enhanced Mannequin). The mini-VERM consists of a Microsoft Kinect motion sensing device and an audio-visual feedback system. This setting provides a real-time feedback about the compression rate, compression depth etc. to the person performing CPR compressions. This was perceived to be an easy to use self-training mechanism for the health care professionals and others to help them improve the quality of the CPR.

In addition to CPR activities, other medical activity detection systems include gesture detection, or detection of a particular medical procedure. In [51], an activity recognition system for trauma resuscitation procedure is outlined where attributes from different image regions and components are fused and decisions about each activity are made using

temporal reasoning. Low level features are used to represent visually salient procedures. The medical procedure is inferred by combining attribute probabilities with logical semantics using an efficient Markov Logic Network Model. An automated transcription of trauma resuscitation in emergency department is presented which is able to track and transcribe the medical procedures performed during resuscitation of a patient, the time instances of their initiation and their temporal durations. Trauma resuscitation is a challenging multi-agent and multi-task setting, in which procedures need to be detected and recognized in a continuous video stream. In [94] Kinect was used to capture the motion of a user to allow a touchless manipulation of the operation table. The gesture dynamics are synchronized with the table movements. However, these methods rely on Kinect motion sensing device and specifically developed software to provide audiovisual feedback.

Several works have also been done using the LBP feature and its variants in the medical field. For instance, in [95] Sarvvinda et al proposed an improvement of the standard LBP, for 2D and 3D analysis of brain image texture for the diagnosis of Alzheimers disease. This method utilizes sign and magnitude features extracted from the axial, coronal an sagittal areas of the brain to create the new Advanced Local Binary Patten Sign Magnitude (ALBPSM) feature vector. In [96] the authors present a fuzzy local binary pattern (FLBP) operator to encode the spatio-temporal characteristics of human gait sequence. Gabor filters are applied on gait energy images for one gait cycle and FLBP is used to encode the resulting Gabor image. Rizwan et.al proposed a method for detecting pain in videos [97] by using pyramid histogram of orientation gradients (PHOG) and pyramid local binary patterns (PLBP) to get a discriminate representation of face. This model is said to achieve real-time pain monitoring without any human intervention.

In [40], a method is proposed to validate individual human actions in the operations of a home medical device to see if the patient has correctly performed the required actions in the prescribed sequence. They used the *MoSIFT* [98] algorithm which first applies the SIFT algorithm to find visually distinctive components in the spatial domain and detects spatio-temporal interest points through (temporal) motion constraints. The motion constraint

consists of a 'sufficient' amount of optical flow around the distinctive points.

The prompt recognition of seizures in new born by the nursery personnel is very important to the early diagnosis and treatment of underlying problems. In [99], an automated procedure for tracking multiple body parts in video recordings of neonatal seizures is presented. Motion in consecutive frames is detected using optical flow and the moving body parts are tracked using Kalman filter.

2.3 Classification Algorithms

Different classifiers have different detection capabilities. Since we deal with high dimensional features, it is not trivial to explain the classifier's response characteristics. We review 3 different classifiers in this section namely, Support Vector Machines, K- Nearest Neighbors and Artificial Neural Networks, in this section.

2.3.1 Support Vector Machines

Support Vector Machines (SVM) [113] are supervised learning models based on learning algorithms that maximize the margin between the training patterns and the decision boundary [113]. The classification function essentially depends only on the *supporting patterns* which are those training examples that are closest to the decision boundary. They are usually a small subset of the training data. Given the set of labeled training data, the SVM algorithm builds a model, that assigns unseen test data into one category or the other, making it a non-probabilistic binary linear classifier. The effective number of parameters is automatically adjusted depending on the complexity of the problem. The solution or model is expressed as a linear combination of the supporting patterns. Thus, the SVM classifier constructs a hyperplane or set of hyper-planes in a high-dimensional space which can be used for classification, regression, and other tasks.

The SVM algorithm uses a kernel, $k(x, y)$, which is selected to suit the problem domain, for its implementation. The hyperplanes in the high dimensional space are defined as the set of points whose dot product with a vector in that space is constant. The learning

of the hyperplane is done by transforming the problem using linear algebra. The SVM kernel can be linear, polynomial or radial.

2.3.1.1 Linear Kernel SVM

For linear kernel, the equation for prediction of a new input is made using the dot product between the input x and each support vector x_i.

$$k(x, x_i) = \sum \alpha_i * (x, x_i) \tag{2.1}$$

where the parameters α_i must be estimated from the training data by the learning algorithm. The kernel defines the similarity measure between new data and the support vectors. The dot product is used as the similarity measure for linear kernel because the distance is a linear combination of the inputs.

2.3.1.2 Polynomial Kernel SVM

In polynomial kernel SVM, we use a polynomial function instead of a dot product. It can be expressed using the equation 2.2.

$$k(x, x_i) = \sum (\alpha_i * (x, x_i))^d \tag{2.2}$$

where d is the degree of the polynomial and must be specified to the learning algorithm. When $d = 1$, this becomes same as linear kernel. The polynomial kernel allows non-linear curved lines in the input space.

2.3.1.3 Radial Kernel SVM

Radial kernel uses more complex radial or exponential kernel function given by equation 2.3.

$$k(x, x_i) = e^{(-\gamma(x-x_i)^2)} \tag{2.3}$$

where γ is a parameter and must be specified to the learning algorithm. The value of γ is often between 0 and 1. The radial kernel is very local and can create complex regions within the feature space, like closed polygons in two-dimensional space.

2.3.2 K-Nearest Neighbors

K-Nearest Neighbors (KNN) [114] algorithm is a non-parametric method of classification. It is an example of instance based learning where the function is only approximated locally and the computation is carried out only during classification. The training phase consists of storing the multidimensional feature vectors and the class labels of the training samples. Classification is done by assigning the label which is most frequent among the k training samples which are nearest to the query point. Here, k is a user defined constant and its value depends on the data. The value of k can be determined by various heuristics such as cross validation. KNN classifiers have been used extensively in activity detection [115, 116].

2.3.3 Artificial Neural Networks

Artificial Neural network (ANN) [117, 118] is another popular supervised learning paradigm based on a collection of connected units called *nodes* or *neurons*. Each connection is capable of transmitting signals between them. The neural network typically has weights between connections that adjust as the learning proceeds. ANN based classification is a powerful and flexible approach and has been used successfully in a variety of action detection tasks [119].

2.4 Fusion of the Outputs of Multiple Detection Algorithms

Pattern recognition algorithms aim at maximizing the classification accuracy. The performance of any classifier is dependent upon several factors such as the amount of available data, the dimensionality of the feature space, and the inter-class separability. For challenging applications, no single classifier can capture all the variation in the data. In

this case, multiple classifiers are used and their results are combined with the help of fusion algorithms. Several fusion methods have been developed to fuse the outputs of multiple classifiers [100]. The main fusion strategies can be broadly classified as (1) data level fusion or low level fusion; (2) feature level fusion or intermediate level fusion; and (3) decision level fusion or high level fusion [101]. In the data level fusion method, raw data from several input sources are fused to obtain new raw data. Feature level fusion combines different features from different modalities to create new feature vector. Decision level fusion combines the outputs from different classifiers using different strategies to obtain a single confidence value. A general architecture for information fusion is illustrated in figure 2.11. It shows how fusion techniques applied at different levels lead to different specific models for expert combination. Decision level fusion combines decisions coming from several experts. By extension, one speaks of decision fusion even if the experts return a confidence (score) and not a decision. In our project, decision level fusion is adopted to improve the accuracy of our approach The following subsection outlines some of the decision level fusion strategies that are relevant to our work.

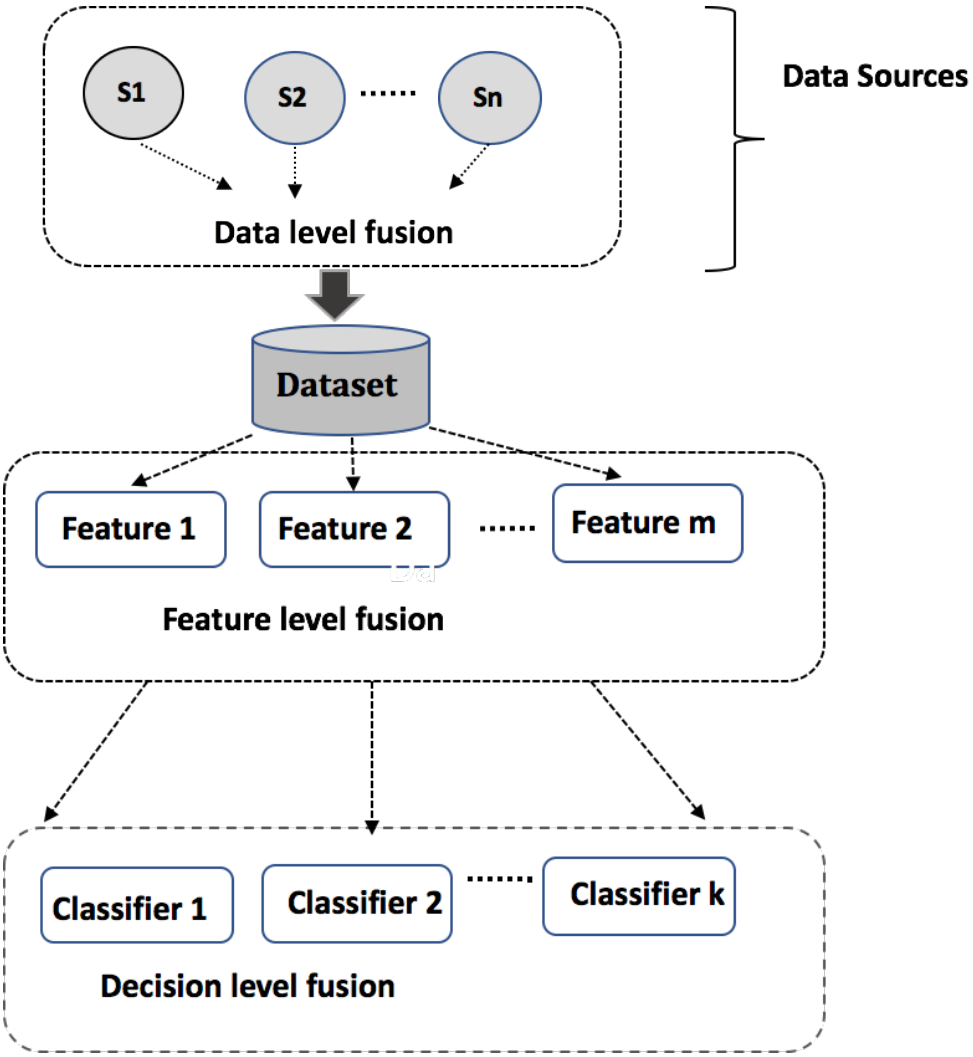


Figure 2.11: General architecture for information fusion

2.4.1 Decision Level Fusion

Often a single classifier is inadequate to handle and correctly represent the variability of data, thereby resulting in reduced accuracy. Typically, different classifiers have different weaknesses and produce different mistakes. Similarly, different classifiers can have different strengths and thereby exhibit different detection capabilities. Decision level fusion of different classifiers is the most common method of combining the detection powers of different classifiers to improve the overall detection accuracy. It has been applied to various fields including character recognition [102], speech recognition [103] and text categorization [104],

and has been proved to be superior to single classifier systems. In our work, we experiment with the three methods outlined in the following subsections.

2.4.2 Ranking

A common approach for fusing of multiple classifier outputs is based on Ranking, such as the Borda Count method [105]. This method first ranks the output of each algorithm, and then sums the individual ranks to generate the combined ranks. It is a single winner election method where each voter votes for their preferred candidate. The magnitude of Borda count is a measure of the strength of agreement by the participating classifiers that the input pattern belongs to that class. The Borda count determines the winner of the election by giving each candidate a certain number of points which corresponds to the position in which the candidate is ranked by the voter. Formally, Borda count, B_j, for class j is the sum of the number of classes ranked below class j by each classifier, i.e.

$$B_j = \sum_{i=1}^{m} B_i(j)) \tag{2.4}$$

In 2.4, $B_i(j)$ is the number of classes ranked below class j by classifier i and m is the number of classifiers. Borda count method does not require any training and is based on the assumption of additive independence among contributing classifiers. The main weakness of this method is that it treats all classifiers equally and does not take into consideration the individual classifier capabilities.

2.4.3 Logistic Regression

The Logistic Regression [105] is a generalization of the Borda Count method, where the weighted sum of individual ranks is calculated and the weights are determined by logistic regression. This method takes into account the relative significance of the classifiers from a combination perspective. The logistic regression method tries to distinguish between the classification correctness and the classifier correlation by treating them as separate problems to be modeled. Let $y_1, y_2, ..., y_m$ be the output or rank scores assigned by m different

classifiers. The logistic response function that combines the multiple series is defined as:

$$\pi(y) = \frac{exp(\alpha + \beta_1 y_1 + \beta_2 y_2 + ... + \beta_m y_m)}{1 + exp(\alpha + \beta_1 y_1 + \beta_2 y_2 + ... + \beta_m y_m)} \tag{2.5}$$

where $\alpha, \beta_1, \beta_2, ..\beta_m$ are constants. The fusion confidence is then calculated using

$$L(y) = log\frac{\pi(y)}{(1 - \pi(y))} = \alpha + \beta_1 y_1 + \beta_2 y_2 + ... + \beta_m y_m \tag{2.6}$$

In 2.6, the transformation $L(y)$ is called *logit*, which is linearly related to y and provides a new value according to which combined rankings are created. The model parameters $\alpha, \beta_1, \beta_2, ..\beta_m$ are estimated using data fitting methods based on maximum likelihood. The relative magnitudes of parameters indicate the relative significance of classifiers in their marginal contribution towards the *logit*.

The estimated model is used to predict the logit for each class, for every test pattern. The classes can simply be sorted by the predicted logits in descending order, for ranking purpose. The class with the largest logit is then considered as most likely to be the true class. The values of $\pi(y)$ or the logit can also be used as a confidence measure. A threshold value is determined empirically and the classes with confidence value below it are rejected.

2.4.4 Discriminant Analysis

Discriminant analysis is similar to regression analysis and is used to determine which predictor variables are related to the dependent variable and to predict the value of the dependent variable, given certain values of the predictor variable. It assumes that different classes generate data based on different Gaussian distributions [106, 107]. The model estimates the mean and variances from the data for each class. The predictions are made by estimating the probability that a new set of inputs belong to each class. The model used Bayes' theorem to estimate the probabilities. Let $y_1, y_2, ..., y_m$ be the output of m different classifiers, and β are the linear model coefficients, the score function Z is defined as

$$Z = \beta_1 y_1 + \beta_2 y_2 + ... + \beta_m y_m \tag{2.7}$$

Suppose two classes of observations have means μ_1 and μ_2, and variance σ, then the linear combination of features represented by equation 2.7 will have means $\beta^T\mu_1$ and $\beta^T\mu_2$ and variances $\beta^T\sigma\beta$ for $i = 1, 2$. Fisher in [108] defined the separation between these two distributions to be the ratio of variance between classes to the variance within classes, i.e.

$$S(\beta) = \frac{\beta^T\mu_1 + \beta^T\mu_2}{\beta^T\sigma\beta} \tag{2.8}$$

which can be represented as

$$S(\beta) = \frac{\bar{Z}_1 - \bar{Z}_2}{\text{Variance of Z within groups}} \tag{2.9}$$

Given the score function, the problem is to estimate the linear coefficients that maximize the score which can be solved by the following equations. Given the mean vectors, μ_1 and μ_2 and the covariance matrices, σ_1 and σ_2 and the number of observations in the respective classes n_1, n_2, we can estimate β and σ using

$$\beta = \sigma^{-1}(\mu_1 - \mu_2) \tag{2.10}$$

$$\sigma = \frac{1}{n_1 + n_2}(n_1\sigma_1 + n_2\sigma_2) \tag{2.11}$$

Classification is done by projecting onto the maximally separating direction, and classifying it as C_1 or C_2 if:

$$\beta^T(y - (\frac{\mu_1 + \mu_2}{2})) > -log\frac{p(C_1)}{p(C_2)} \tag{2.12}$$

The main difference between discriminant analysis and logistic regression is that, for discriminant analysis, the dependent variable must be categorical and independent variables must be continuous in nature.

CHAPTER 3

CPR SCENE RETRIEVAL FRAMEWORK BASED ON OBJECT AND ACTIVITY DETECTION

In this chapter we elucidate our proposed framework to detect CPR activity from medical simulation videos, retrieve the scenes containing CPR activity, and to evaluate the correctness of the performed CPR procedure. Medical simulation videos are recorded by the supervising physicians for educational purposes. Then, they are evaluated, and the results are used to debrief the trainees about their performance during the simulated emergency scenario. The objective is to provide the physician with tools that can *"query all the scenes that contain CPR activity"* and *"return all scenes that contain CPR and ventilation given in the correct/incorrect ratio"*. First, we present the framework for temporal segmentation of the input video into overlapping video volumes. Then we illustrate how we select the regions of interest for detecting the CPR activity. Next, we describe the spatio-temporal shape and texture feature extraction processes. These features will be used to encode video simulations and learn a binary classifier, that can discriminate between CPR and non-CPR scenes. Finally, we explain the breathing bag detection procedure. An overview of the proposed system is illustrated in figure 3.1. The different steps of our proposed framework are described in the following sections.

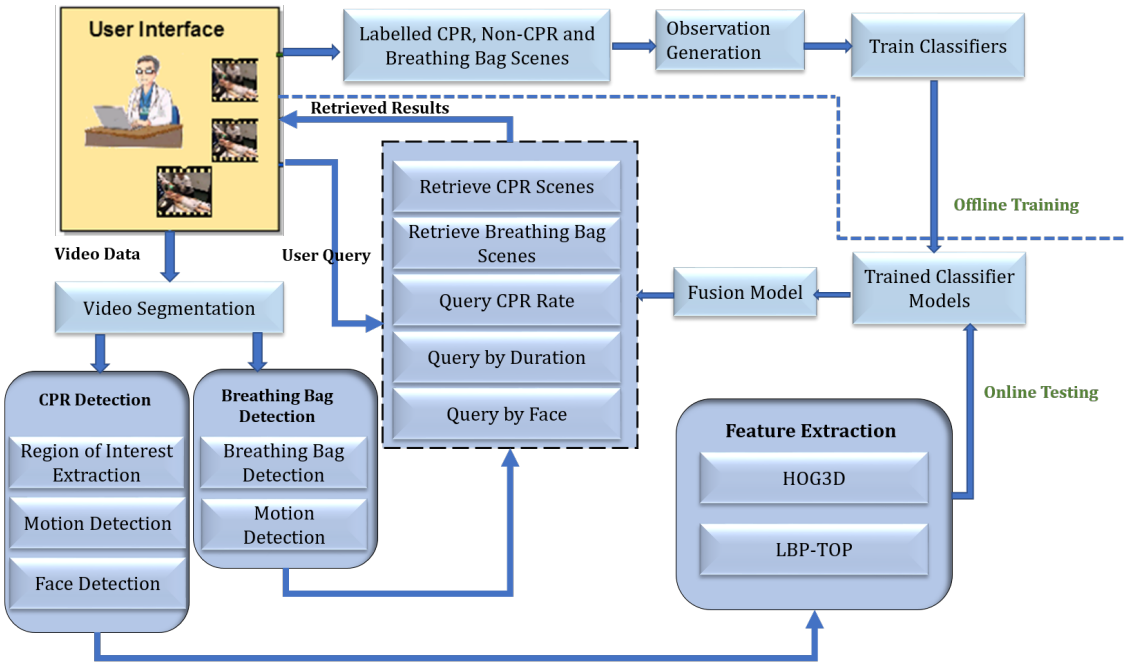


Figure 3.1: Overview of the proposed framework

3.1 CPR Activity Detection

The proposed system for for CPR activity detection,is composed of three main components: 1) Video Segmentation; 2) Pre-screener module to detect region of interest and presence of activity and; 3) Classification module that extracts features from the detected activity regions and classifies them to be CPR or non-CPR activity regions. These components are described in the following subsections.

3.1.1 Video Segmentation

CPR activities are localized within a small spatial region and over a short temporal duration. The features should be extracted from a small region of interest to get a good description and representation for the CPR activity. In order to choose the best value for temporal overlap, we rely first on the optical flow in order to have a clear understanding about the motion around the hand of the person performing the CPR activity. We use the Horn-Schunck method [109] to calculate the optical flow.

The Horn-Schunck algorithm tries to minimize distortions in flow and assumes smoothness in flow over the whole image. The flow is formulated as a global energy function which is sought to be minimized. For two-dimensional image streams, this function is defined as:

$$E = \int\int [(I_x u + I_y v + I_t)^2 + \alpha^2(\|\Delta u\|^2 + \|\Delta v\|^2)]dxdy \tag{3.1}$$

where I_x , I_y and I_t are the derivatives of the image intensity values along the x, y and t dimensions respectively, $\vec{V} = [u(x,y), v(x,y)]^T$ is the optical flow vector, and the parameter α is a regularization constant. Larger values of α lead to a smoother flow. The function in (3.1) can be minimized by solving the associated multi-dimensional Euler-Lagrange equations:

$$\begin{cases} \frac{\partial L}{\partial u} - \frac{\partial}{\partial x}\frac{\partial L}{\partial u_x} - \frac{\partial}{\partial y}\frac{\partial L}{\partial u_y} = 0 \\ \frac{\partial L}{\partial v} - \frac{\partial}{\partial x}\frac{\partial L}{\partial v_x} - \frac{\partial}{\partial y}\frac{\partial L}{\partial v_y} = 0 \end{cases} \tag{3.2}$$

where L is the integrand of the energy expression, giving

$$\begin{cases} I_x(I_x u + I_y v + I_t) - \alpha^2 \Delta u = 0 \\ I_y(I_x u + I_y v + I_t) - \alpha^2 \Delta v = 0 \end{cases} \tag{3.3}$$

In 3.3, $\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ is the Laplace operator. In practice, the Laplacian is approximated numerically using finite differences, and may be written as $\Delta u(x,y) = \bar{u}(x,y) - u(x,y)$, where $\bar{u}(x,y)$ is a weighted average of u calculated in a neighborhood around the pixel at location (x,y). Using this notation, equations in (3.3) may be written as:

$$\begin{cases} (I_x^2 + \alpha^2)u + I_x I_y v = \alpha^2 \bar{u} - I_x I_t \\ I_x I_y u + I_y^2 + \alpha^2)v = \alpha^2 \bar{v} - I_y I_t \end{cases} \tag{3.4}$$

The set of equations in (3.4) is linear in u and v and may be solved for each pixel in the image. However, the solution depends on the neighboring values of the flow field and hence must be repeated once the neighbors have been updated. The following iterative scheme is used for this:

$$\begin{cases} u^{k+1} = \bar{u}^k - \frac{I_x(I_x\bar{u}^k + I_y\bar{v}^k + I_t)}{\alpha^2 + I_x^2 + I_y^2} \\ v^{k+1} = \bar{v}^k - \frac{I_y(I_x\bar{u}^k + I_y\bar{v}^k + I_t)}{\alpha^2 + I_x^2 + I_y^2} \end{cases} \tag{3.5}$$

In 3.5, the superscript k stands for iteration number. So, as we can notice, optical flow measures the change in the velocity in terms of speed and direction at each pixel location. To get an insight into the temporal extend of a CPR action cycle, we manually extract region around the hands from CPR scenes that are labeled manually and we compute the optical flow for all pixels within it. A single optical flow vector extracted for the region is then estimated as the average of the optical flow of all if its pixels. Figure 3.2 displays the average optical flow of a sample CPR activity region.

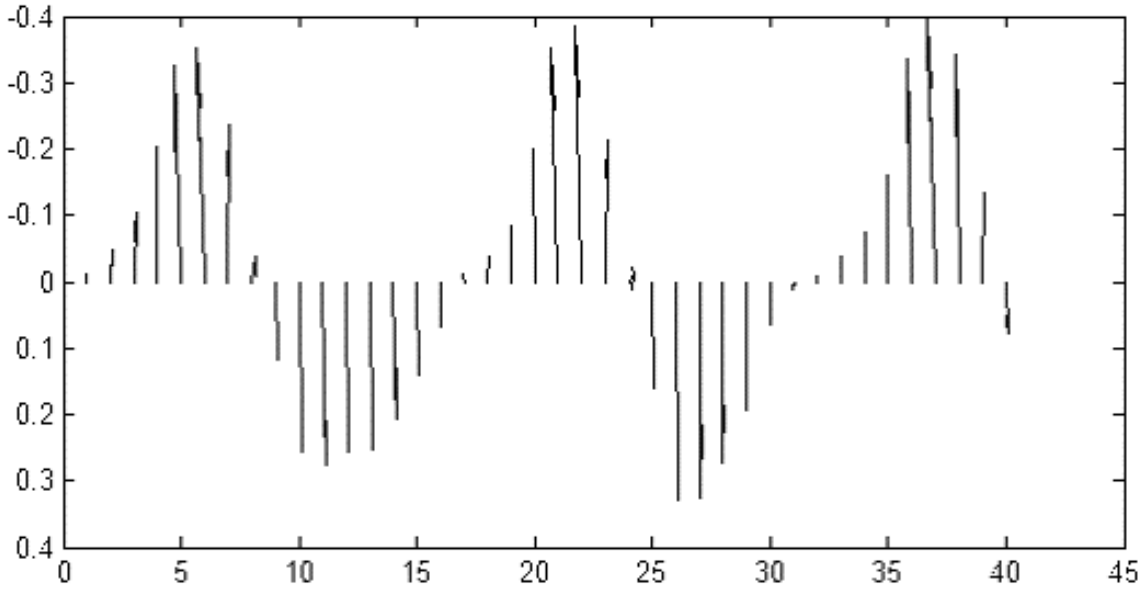


Figure 3.2: Average optical flow of a sample CPR sequence

After analyzing the average optical flow for many CPR scenes and using our knowledge about the typical frequency of CPR and the temporal resolution of the record video, we set the number of frames for each volume of training and testing to be 18 frames. A sequence of 18 frames (approximately 0.6 seconds) of CPR action typically corresponds to one CPR cycle (up-down movement) which is essential for capturing the rhythmic cycle of the CPR activity. Therefore, every spatial region of interest will be segmented into overlapping

volumes that include 18 frames.

3.1.2 Pre-screener Module

The medical simulation videos are recorded in an emergency room where a medical condition is simulated in the dummy. An emergency room scene involves multiple persons and multiple actions. The up-down hand movement can happen almost anywhere and anytime on the scene, and not restricted to CPR activities. Also, there could be scenes with absolute inactivity. Therefore, we employ a pre-screener module to select the region of interest as well as identifying and ignoring the scenes with absolute inactivity.

3.1.2.1 Extraction of Region of Interest

A CPR activity is typically characterized by the trainee's hands being placed on the center of the chest of the dummy. So, in order to reduce the computational complexity of our approach, we restrict our processing to those regions of interest. We start by detecting the chest region of the dummy. All subsequent processing steps will be applied only to these detected regions of interest. Since in all medical simulation videos, the dummy has a bare chest, we can use a simple skin detector to identify the chest region of the dummy. If the chest of the dummy cannot be detected with a high confidence value (due to occlusion), we will scan the entire scene with a moving window volume for motion detection.

The bare chest can be detected using a simple but efficient skin pixel classifier to discriminate between skin and non-skin regions. First, we collect several skin regions under different illuminations and orientations, from our database of images as shown in figure 3.3. Each sample is then mapped to the $YCbCr$ color space, where Y stands for luminance, Cb for chrominance blue and Cr stands for chrominance red. $YCbCr$ is better suitable for representing skin-color than the common RGB representation. We only use the Cb and Cr components of the color to reduce the effect of noise. The color distribution from all the skin samples is used to fit a Gaussian model with mean μ and covariance C. The probability density function that best fits the data is then modeled using:

Figure 3.3: Skin sample collection from dummy

$$p(x|\mu, C) = \frac{1}{(2\pi)^{\frac{d}{2}}(det(C)^{\frac{1}{2}})} exp(-\frac{1}{2}(x-\mu)C^{-1}(x-u)^{t}) \tag{3.6}$$

To identify the chest region in any test image, the likelihood of each pixel is computed using the Gaussian function in equation (3.6).The resulting skin likelihood map is then passed through a low pass filter for smoothing. Finally, the image region with the maximum likelihood score is identified as the chest region. This process is illustrated in figure 3.4 . Here, the likelihood score of a region is the sum of likelihood of each pixel within the region. After identifying the chest region, we proceed to the motion detection step.

3.1.2.2 Motion Detection

The simulation videos may have several scenes of inactivity. To improve the efficiency of our approach, we limit the processing to scenes that involve some motion. Therefore, the first step is to identify the scenes with any activity. To achieve this task, we use a simple motion detection method.

Using Q consecutive image frames, $I(t), t = 1...Q$, first, we calculate the mean image

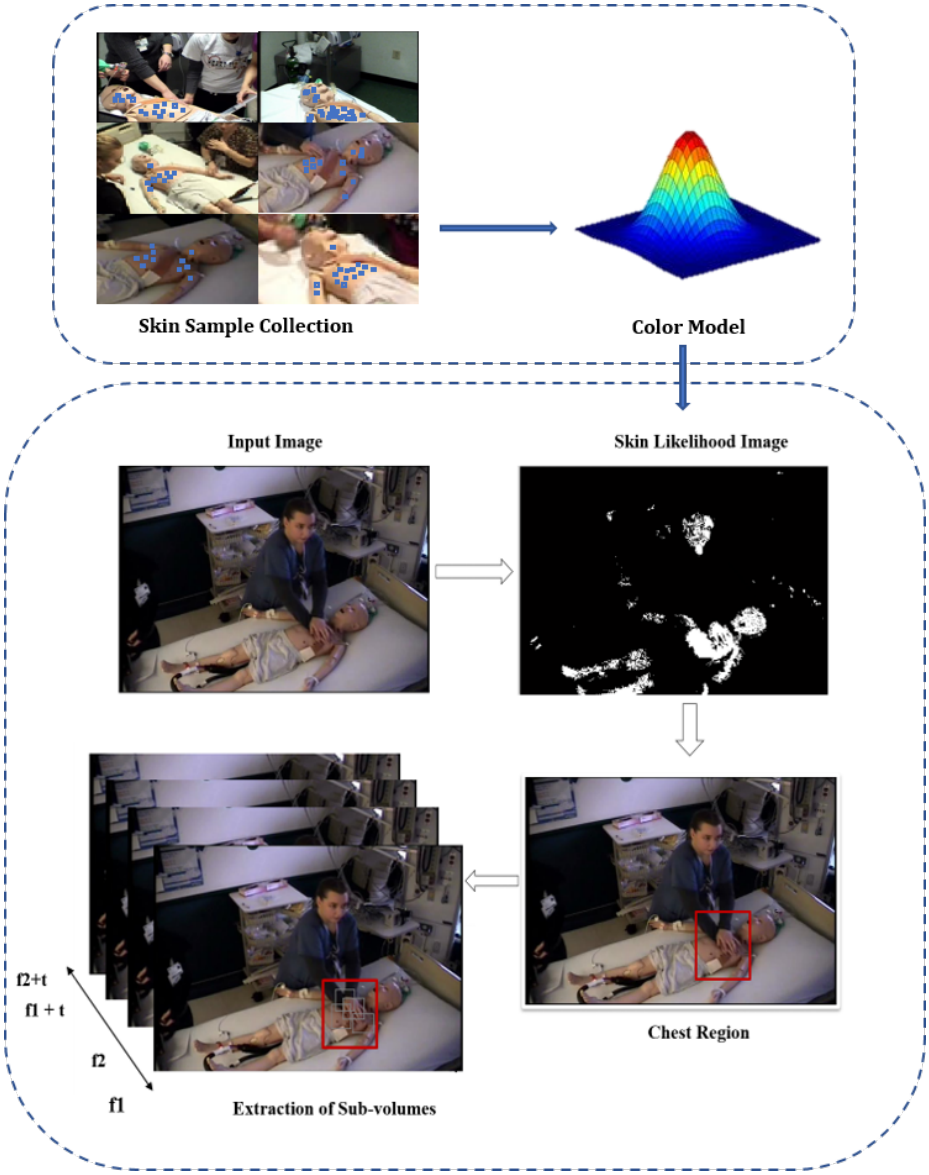


Figure 3.4: ROI Selection Process

B by averaging the corresponding pixels, that is,

$$B(x,y) = \sum_{t=1}^{Q} \frac{I_{(x,y)}(t)}{Q} \tag{3.7}$$

After calculating the average image B, we subtract it from the first image $I(1)$ and threshold it. The resulting binary image $V(x,y)$ is defined as:

$$V(x,y) = \begin{cases} 1, & if|I_{(x,y)}(1)) - B(x,y)| \geq Th \\ 0, & otherwise \end{cases} \tag{3.8}$$

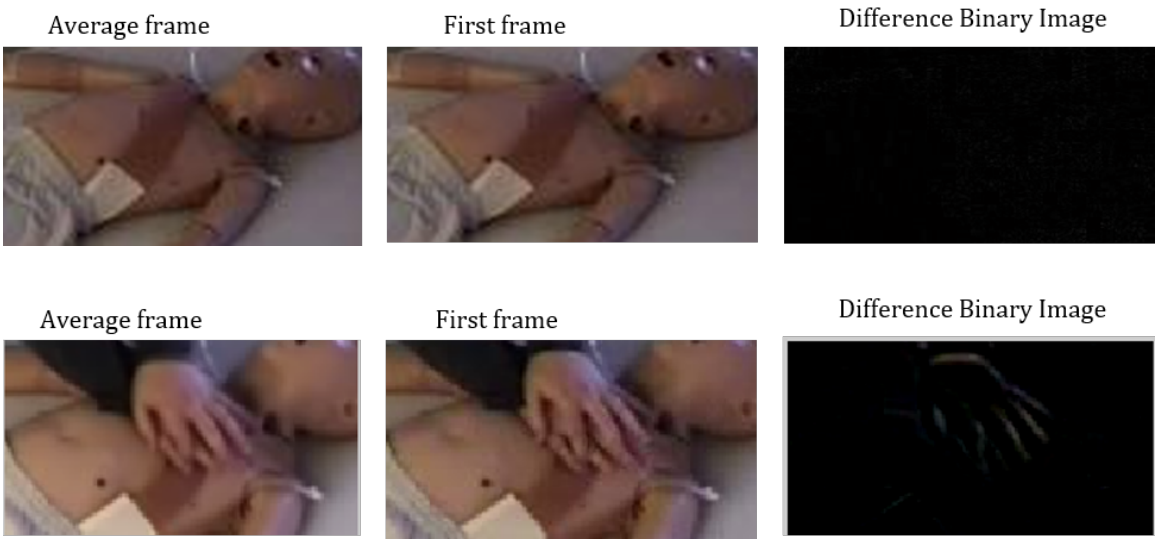


Figure 3.5: Binary images in the absence (top) and presence of motion (bottom)

Figure 3.5 top shows an example of binary image obtained for $Q = 18$ images of interest picked from a CPR scene and figure 3.5, bottom shows an example of binary image obtained for $Q = 18$ images which has no motion. If the sum of pixel values in V exceeds a threshold, we identify it as occurrence of motion.

3.1.3 Classification of the Identified Activity Regions

We use spatio-temporal features to classify the detected activity regions into CPR and non-CPR classes. The advantage of using spatio-temporal features is that the variations in the space and time dimensions can be represented using a single feature vector. The objective of our proposal is to identify CPR activity and breathing bag activity scenes without video shot segmentation or tracking. We show that this is possible with the use of features that represent the spatio-temporal shape of the activity (HOG3D) and, spatio-temporal texture of the activity (LBP-TOP). These features are described in the following subsections.

3.1.3.1 Spatio-temporal Histogram of Oriented Gradients (HOG3D)

The HOG3D [3] is an efficient descriptor to represent the pixel intensity variations in spatial and temporal dimensions. It jointly encodes both appearance and motion information. The HOG3D descriptors are based on spatio-temporal gradient orientations, and hence they are robust to changes in illumination and minor deformations. While performing a CPR activity, the hands of the actor will typically be in one specific posture as shown in figure 3.6,(a). In our proposed work, we use the HOG3D features to capture the space-time orientation of gradients that represents the structure of the CPR action cycle. The HOG3D features are computed as described below.

We divide each detected activity volume $\mathbf{c} = (x_c, y_c, t_c, w_c, h_c, l_c)^T$ where $(x, y, t)^T$ denotes the position of the center of the volume, and w, h and l its width, height and length respectively, into cuboids of size M x M x N as shown in figure 3.6 (a). This will create the set of M x M x N sub-blocks, $\mathbf{b_j}$, each of size S x S x S.

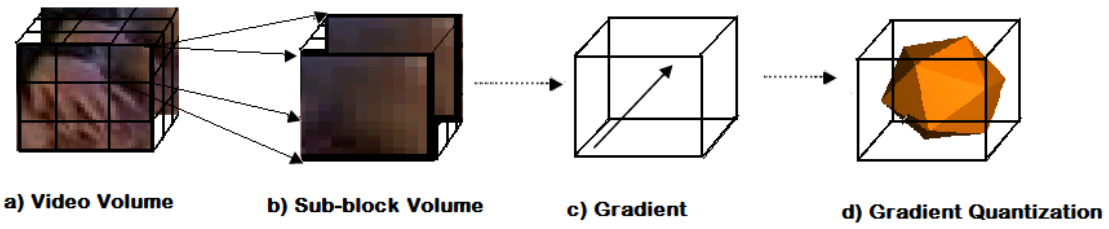


Figure 3.6: Steps for computing HOG3D features

For each b_j, first we compute the gradients along the x, y, and t directions at every pixel. Then, for each pixel, the gradient orientation is quantized by projecting the (dx, dy, dt) vector on a 20 dimensional regular polygon (or icosahedron), with the gradient magnitude as its weight. We will refer to the quantized block as $\mathbf{q_{b_j}}$. Then, for each $\mathbf{q_{b_j}}$, the weighted gradients are smoothed using a 3-D Gaussian filter, with σ determined by the size of $\mathbf{b_j}$. The resulting vector is then thresholded and normalized. The histogram $\mathbf{h_c}$ for the region $\mathbf{c}$ is constructed by accumulating the quantized gradients $\mathbf{q_{b_j}}$ across all S^3 voxels within b_j.

$$h_c = \sum_{i=1}^{S^3} q_{b_i} \tag{3.9}$$

Each h_c is then normalized using $L2$ norm within each sub-block. These histograms are finally concatenated to form the HOG3D descriptor of M x M x N x 20 dimensions. For example, if we select $M = 2$, and $N = 3$, then the resulting feature vector will have $2 \times 2 \times 3 \times 20 = 240$ dimensions.

3.1.3.2 Local Binary Patterns over Three Orthogonal Planes (LBP-TOP)

Dynamic Texture is the extension of standard texture features to the temporal domain. The LBP-TOP feature combines the motion and appearance information of the underlying texture. In a typical CPR action cycle, the dynamic texture is expected to display a similar pattern for different actors. Moreover, the texture features in a small local neighborhood of the volume of interest are not only insensitive to the translation and rotation, but also robust with respect to the illumination changes.

In order to capture the dynamic texture of the CPR cyclic activity we use the LBP-TOP feature. The LBP-TOP is a reduced representation of VLBP (as described in 2.1.1), where the local binary patterns are computed only for the three orthogonal planes XY, XT and YT. The XT and YT planes contain the temporal information in the volume. The three orthogonal planes intersect at the center voxel. The features are extracted from each plane and concatenated to form the final feature histogram which serves as a global descriptor for the spatial and temporal features. These steps are illustrated in figure 3.7.

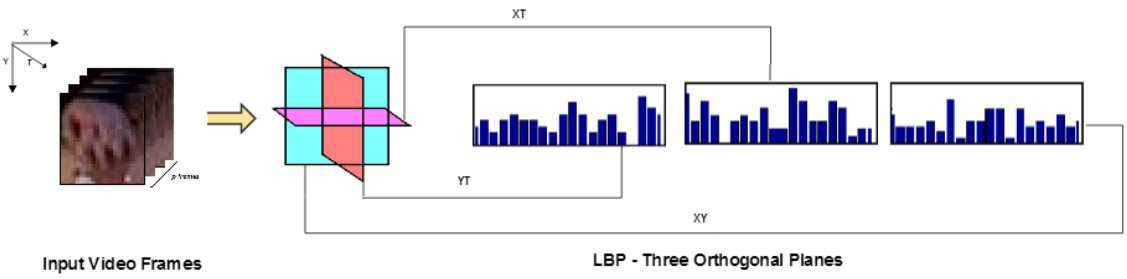


Figure 3.7: Steps for computing LBP-TOP features

For LBP-TOP features, as outlined in chapter 2, the number of neighborhood points P determines the size of the feature vector. A neighborhood of P points will result in 2^P codes. So, the LBP-TOP feature vector will be a 3×2^P dimensional histogram. The radius in the time axis can be the same as, or different from that of the radii in the space axis, depending on the extend of variation of texture in the time and space axis. This will sometimes result in using an elliptical neighborhood instead of the conventional circular neighborhood. Let the radii in the X, Y and Z axis be R_X, R_Y, and R_z, and the number of neighboring points in the XY, XT, and YT planes be P_{XY}, P_{XT} and P_{YT}. If the coordinates of the center pixel of the volume of interest are (x_c, y_c, t_c), then the coordinates of the neighboring points in the XY plane are given by $(x_c - R_X sin(2\pi p/P_{XY}), y_c + R_Y cos(2\pi p/P_{XY}, t_c)$. Similarly, the coordinates of neighboring points in the XT plane are given by $(x_c - R_X sin(2\pi p/P_{XT}), y_c, t_c - R_T cos(2\pi p/P_{XT})$, and the coordinates of the neighboring points in the YT plane are given by $(x_c, y_c - R_Y sin(2\pi p/P_{YT}), t_c - R_T cos(2\pi p/P_{YT})$. For any volume of interest, the LBP-TOP histogram is defined as

$$H_{i,j} = \sum_{x,y,t} I\{f_j(x,y,t) = i\} \text{ for } i = 0, ...n_j - 1;\ j = 0, 1, 2 \tag{3.10}$$

In 3.10 n_j is the number of different labels produced by the LBP operator in the jth plane $(XY, (j = 0), XY, (j = 1) and YT, (j = 2))$, $f_i(x, y, t)$ is expressed as the LBP code of the center pixel (x, y, t), in the jth plane and

$$I\{A\} = \begin{cases} 1, & if\ A\ is\ true \\ 0, & if\ A\ is\ false \end{cases} \tag{3.11}$$

The final histogram is then normalized to get a coherent and generalized feature descriptor.

3.2 Face Detection

We augment the CPR scene retrieval framework with face detection. Often, the physician supervising the simulation session is interested in retrieving the CPR activity scenes, performed by a particular person. Thus in out system we include a component that

provides the physician with necessary tools to enable person specific CPR scene retrieval, that could be a doctor or a nurse or any health care practitioner.

We use the Viola and Jones face detection method [76]. This algorithm can detect faces in real time with very low false alarm rate using Haar-like features trained by the AdaBoost algorithm [110]. We first identify the faces. For this, we run the face detection algorithm in a sampled subset of the video frames. The detected faces are clustered to create our face database. To retrieve the CPR scenes performed by a selected person, we apply the face recognition algorithm. We compare the face detected from the CPR scene with the selected face in the dataset [111, 112], and retrieve the scenes with the corresponding face selection. Figure 3.8 shows an overview of the face detection module.

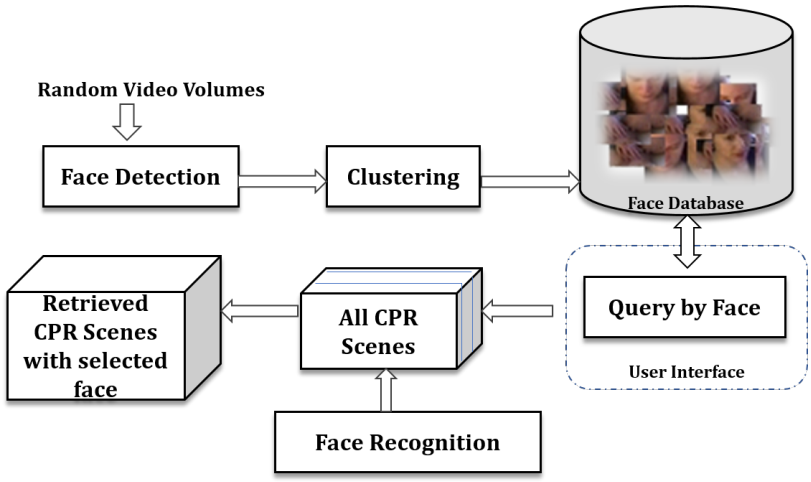


Figure 3.8: Overview of face detection module

3.3 Breathing Bag Activity Detection

The breathing bag activity detection process follows two steps: First, we detect the presence of the breathing bag in the screen and second, we classify the scene with the breathing bag as active or not active. These steps are explained in detail in the following subsections.

3.3.1 Pre-screener for Detecting the Breathing Bag

The breathing bags generally are of a few specific colors. Therefore, color masks can be used effectively for the detection of breathing bags in the scene. The breathing bags are almost always present in all the video frames, either in action or idle. The HSV color space is one of the popular color spaces which separates*luma*, or the image intensity, from *chroma* or the color information, and it corresponds to how people experience color better than RGB color space. As hue varies from 0 to 1.0, the corresponding colors vary from red through yellow, green, cyan, blue, magenta, and back to red, so that there are actually red values both at 0 and 1.0. As saturation varies from 0 to 1.0, the corresponding colors (hues) vary from unsaturated (shades of gray) to fully saturated (no white component). As value, or brightness, varies from 0 to 1.0, the corresponding colors become increasingly brighter. Figure 3.9 illustrates HSV color space.

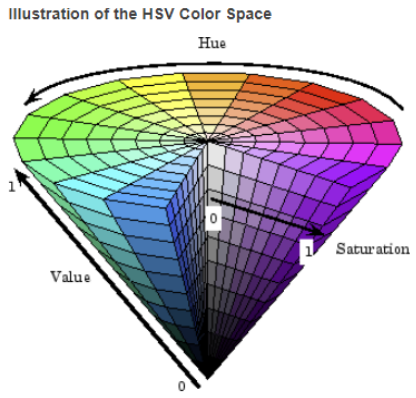


Figure 3.9: Illustration of HSV color space (Matlab)

During the training phase, the color (H, S and V values) of the breathing bag is extracted and recorded for several frames. We use the hue and saturation distribution to fit a Gaussian model with mean μ and covariance matrix C as shown in equation 3.6. To detect the breathing bag during testing, the likelihood of each pixel in the frame is computed using the Gaussian probability density function shown in equation 3.12, where μ is the mean and σ is the variance.

$$P(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2} \tag{3.12}$$

The areas with the maximum likelihood values are considered as potential candidates for the breathing bag. This is further smoothed with a 2D median filter, where each output pixel contains the median value in a 3-by-3 neighborhood around the corresponding pixel in the input image. In a median filter, a window slides along the image, and the median intensity value of the pixels within the window w, becomes the output intensity of the pixel being processed, as shown in equation 3.13. This is useful in preserving edges in the image while reducing random noise.

$$y[m,n] = median\left\{x\left[i,j\right],(i,j)\epsilon w\right\} \tag{3.13}$$

where w represents a neighborhood defined by the user, centered around location $[m,n]$ in the image. Then morphological opening is applied. Morphological opening is defined as an erosion followed by a dilation using the same structuring element for both operations. IF $\ominus$ represents erosion and *oplus* represents dilation, morphological opening of A by B can be given by equation 3.14. The morphological opening smooths the object borders and removes small objects while preserving the shape and size of larger objects in the original binary image.

$$A \circ B = (A \ominus B) \oplus B \tag{3.14}$$

The connected components are found from this image and the blob that has the maximum area within a threshold is selected as the breathing bag. If there are other objects in the scene which has the same or similar color as that of the breathing bag (green in our experiments), we specify the search window, which is determined manually for each video, to reduce false detections. Figure 3.10 shows the overview of the breathing bag detection process.

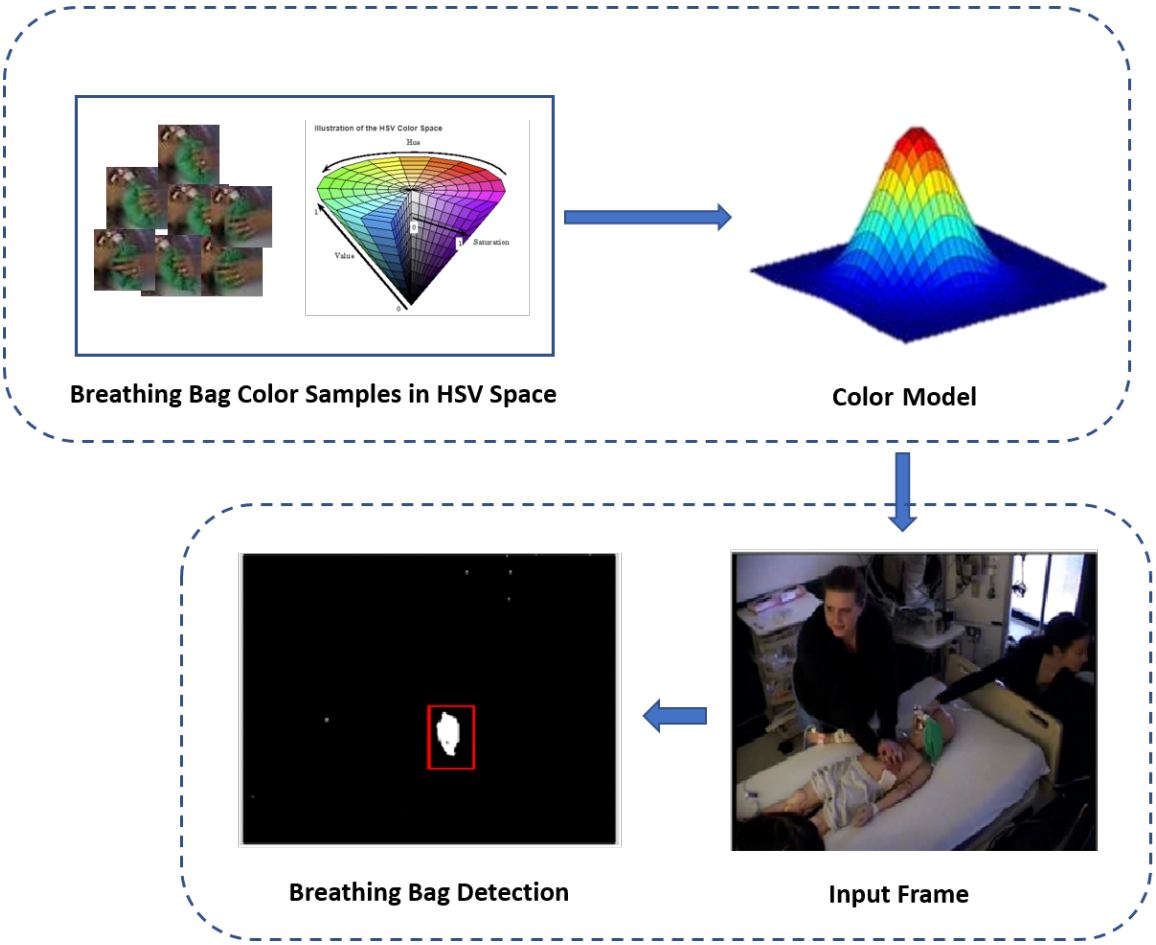


Figure 3.10: Overview of breathing bag detection

3.3.2 Breathing Bag Activity Classification

The breathing bag activity is characterized by the compression and relaxation of the breathing bag. During medical emergency, the breathing bag mask ventilation can be given at any time, not necessarily during the CPR activity. The duration of the breathing bag activity can range anywhere from 13 to 80 frames. During the breathing bag activity, the area of the bag falls abruptly in about 6 to 10 frames. Figure 3.12 shows how the detected area of the breathing bag changes when there is a breathing bag activity. The red region shows the frames where there is breathing bag activity, and the blue plot corresponds to the area of the detected bag by the frame number. We can see that, whenever there is a breathing bag activity, when the bag is squeezed, there is a reduction in its area. We use

this property to detect the frames with the breathing bag activity.

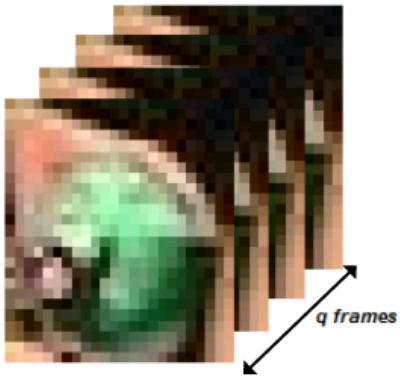


Figure 3.11: Example of a 3D bounding box of breathing bag action sequence

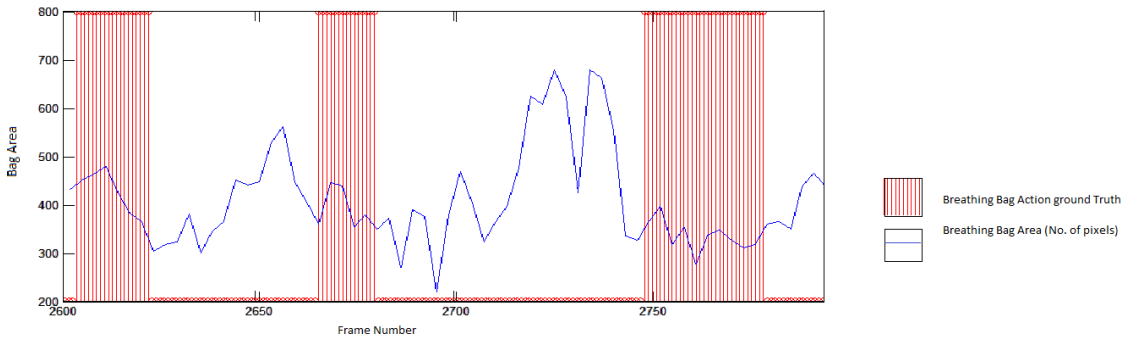


Figure 3.12: Example showing how the breathing bag area changes during breathing bag activity

3.4 Classification

Here, we elaborate the parameter selection and implementation details of the different classifiers used in our experiments. To gain a better understanding of the variation in the data characteristics, we experiment with 3 different classifiers, namely Support Vector Machines, K-Nearest Neighbors and Artificial Neural Networks.

3.4.1 Support Vector Machines

The SVM has proved to perform efficiently in many of the classification tasks [1,3,63], especially when the features have very high dimensions. Since in this work we deal with high dimensional data, we evaluate our framework with the SVM classifier. A simple SVM classifier, with a linear kernel, is used to build the model, with the computed spatio-temporal features for CPR activity and non-CPR activity. Separate SVM models are built for HOG3D and LBP-TOP features.

3.4.2 K-Nearest Neighbors

In KNN classification, the object is assigned to the class most common among its k nearest neighbors. Several distance measures can be used to compute the similarity. In our experiments we use the default euclidean distance to measure the similarity between neighbors.

3.4.3 Artificial Neural Networks

ANN consist of input and output layers, as well as a hidden layer consisting of units that transform the input into values that further serve as inputs to the output layer. The figure 3.13 illustrates this functionality. These hidden states are similar to biological neurons. Each of these hidden state is a transient form with a probabilistic behavior. A grid of such hidden states act as a bridge between the input and the output.

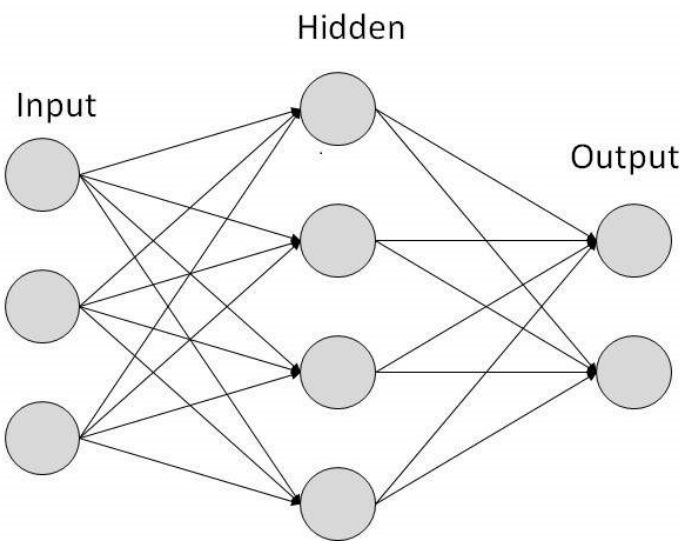


Figure 3.13: Illustration of ANN

In the above diagram, suppose there is a vector of three inputs and the probability that the output event will fall into class 1 or class 2. For this prediction we need to predict a series of hidden classes in between. The three vector inputs, in some combination, predicts the probability of activation of teh four hidden nodes. The probabilistic combination of hidden states 1 to 4 are then used to predict the activation of the output nodes, which finally predicts the outcome.

CHAPTER 4

EXPERIMENTAL RESULTS

In this chapter, we present the experimental results and analysis of the proposed work. An overview of our experimental setup is illustrated in figure 4.1. We explain each of the modules in datail, in the following sections.

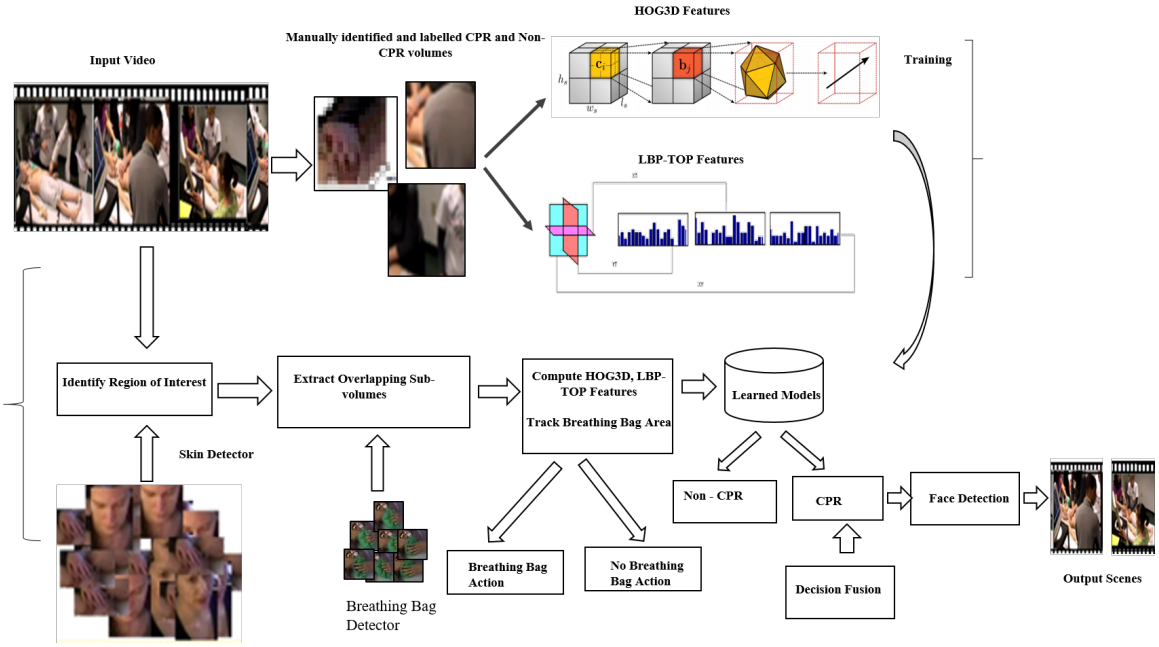


Figure 4.1: Overview of the experimental setup

4.1 Data Collection

The CPR simulation videos are provided by the (SPARC) working group at Kosair Childrens Hospital, Louisville. The duration of these simulation sessions is roughly 15 to 30 minutes, with a temporal resolution of 30 frames per second. Each video has about 4 to 10 actors. The frames have a spatial resolution of 720 x 480 pixels. More details of the data statistics are listed in table 4.1. To provide an insight into the content of the videos, we

TABLE 4.1

Details of the simulation videos used in our experiments

Video	Duration	Number of frames	Number of CPR scenes	Average number of frames per CPR scene	Std. Deviation of number of frames per CPR scene
CPR1	30 m 19 s	54538	29	266.8	123.9
CPR2	16 m 1 s	28831	36	236.5	145.7
CPR3	14 m 57 s	26905	1	7387	0
CPR4	16 m 8 s	29037	46	124.9	110.5
CPR5	17 m 39 s	31751	14	1003.8	1152.6
CPR6	18 m 27 s	33204	56	226	350.9
CPR7	21 m 49 s	39253	28	632.8	750.8
CPR8	15 m 29 s	27865	0	0	0

provide sample frames from eight videos that are used in our experiments, in table 4.2. The video CPR7 involves an infant mannequin, while the other videos have child mannequins.

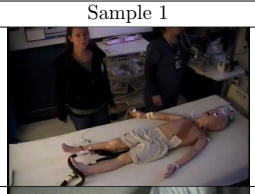
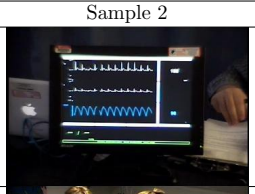
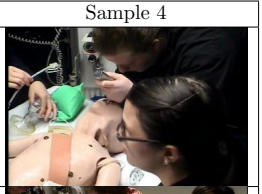
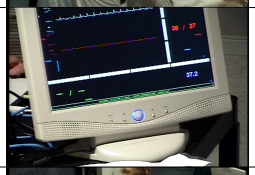
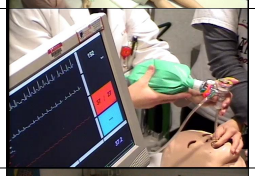
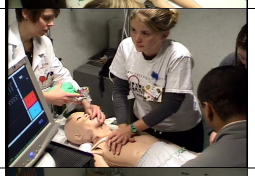
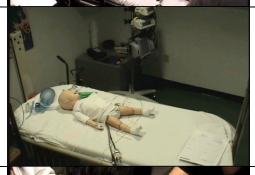
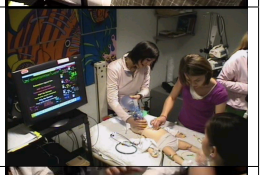
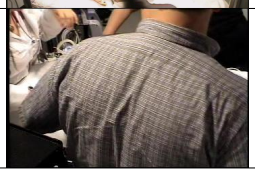
4.2 Analysis of the proposed system using segmented CPR scenes

In the first experiment, we use videos CPR1 and CPR2 to analyze our proposed system with different parameter settings. We notice that different performers have different hand postures while performing the CPR action, and cameras have varied alignments in different videos. CPR1 and CPR2 videos have CPR activity performed by 4 different actors and the camera alignment is different in both the videos. We collect CPR training data from these two videos, so that it correctly represents and captures the variance in the spatio temporal structure of the CPR action, across all videos.

A CPR action cycle, when done correctly (at the rate of 100 compressions per minute) roughly includes 18 frames. Due to the unavailability of the ground truth information, we manually label the CPR frames and non-CPR frames for the experiments. After annotating the CPR and non-CPR scenes, we extract bounding box volumes of CPR action cycles and non-CPR action cycles. After visual inspection of few CPR scenes, the size of the bounding box is fixed as 55×55, so that it covers the spatial extend (XY) of the hands of the actor performing CPR (figure 4.2). Thus, the size of the bounding box volume becomes

TABLE 4.2

Sample frames from medical simulation videos

Video#	Sample 1	Sample 2	Sample 3	Sample 4
CPR1				
CPR2				
CPR3				
CPR4				
CPR5				
CPR6				
CPR7				
CPR8				

$55 \times 55 \times 18$. The non-CPR action cycles are sub-volumes that are randomly selected from anywhere in the scene, that does not enclose a CPR activity. The same bounding box size is used for extracting non-CPR volumes. Videos CPR1 and CPR2 contains contains 29 CPR scenes and 36 CPR scenes respectively as shown in table 4.1. Each CPR scene contains roughly 15 to 40 CPR action cycles. In total, there are 827 CPR cycles in CPR1, and 1080 CPR cycles in CPR2.

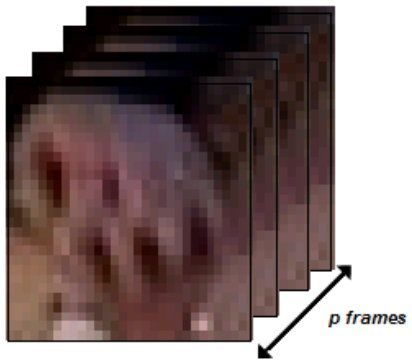


Figure 4.2: Example of a 3D bounding box of a region that correspond to a CPR action sequence

We use one-third of the CPR action cycles from both CPR1 and CPR2, to form the positive training set, Tr^C, which is 650 CPR training samples. We also collect 500 non-CPR action cycles each from CPR1 and CPR2 to represent the negative training set, Tr^N.

4.2.1 SVM model training with HOG3D features

First, we extract HOG3D features from Tr^C and Tr^N. The video volumes are divided into $M \times M \times N$ sub-blocks. We choose $M = 2$ and $N = 2$, which results in a feature vector of $2 \times 2 \times 2 \times 20 = 160$ dimensions. These values were empirically determined to be best suitable for capturing the spatio temporal structure of the hand movement while performing the CPR activity.

Next, we build a linear SVM binary classifier with the extracted CPR and non-CPR

HOG3D features (which we will refer to as SVM-H). In order to prevent overfitting, we determine the regularization parameter C of SVM classifier using LIBSVM [120] functionalities. The value of C was determined to be 2. To evaluate the performance of the features, we use k-fold cross validation. In k-fold cross validation, the original sample is randomly partitioned into k equal sized subsamples. Of the k subsamples, a single subsample is retained as the validation data for testing the model, and the remaining $k1$ subsamples are used as training data. The cross-validation process is then repeated k times (the folds), with each of the k subsamples used exactly once as the validation data. The k results from the folds can then be averaged to produce a single estimation. The advantage of this method over repeated random sub-sampling is that all observations are used for both training and validation, and each observation is used for validation exactly once. The main purpose of using k fold cross validation is to find the best parameters for classifiers with the training data extracted manually and find those which give the best accuracy in order to use them for the second system of scene retrieval.

4.2.2 SVM model training with LBP-TOP features

In the next experiment, we first extract LBP-TOP features fromTr^C and Tr^N. We choose a neighborhood of 8 pixels for the XY, XT, and YT planes. We use elliptical neighborhood instead of the conventional circular neighborhood for the XT and YT planes. The neighborhood radii we used for this experiment was $x_{radius} = y_{radius} = 1$ and $t_{radius} = 2$. We compute the uniform patterns for LBP [121], and construct feature vectors of 177 dimensions. Finally, we train a second linear SVM classifier (which we will refer to as SVM-L) with the new set of features. In practice, SVM tends to be resistant to overfitting because it uses regularization. To avoid over-fitting we carefully tune the regularization parameter, C, which we determine to be equal to 8, using standard LIBSVM tuning functionalities [120]. We first use the train data to determine the value of C, before we do cross-validation. Similar to the previous experiment, we perform 10-fold cross validation with the LBP-TOP features.

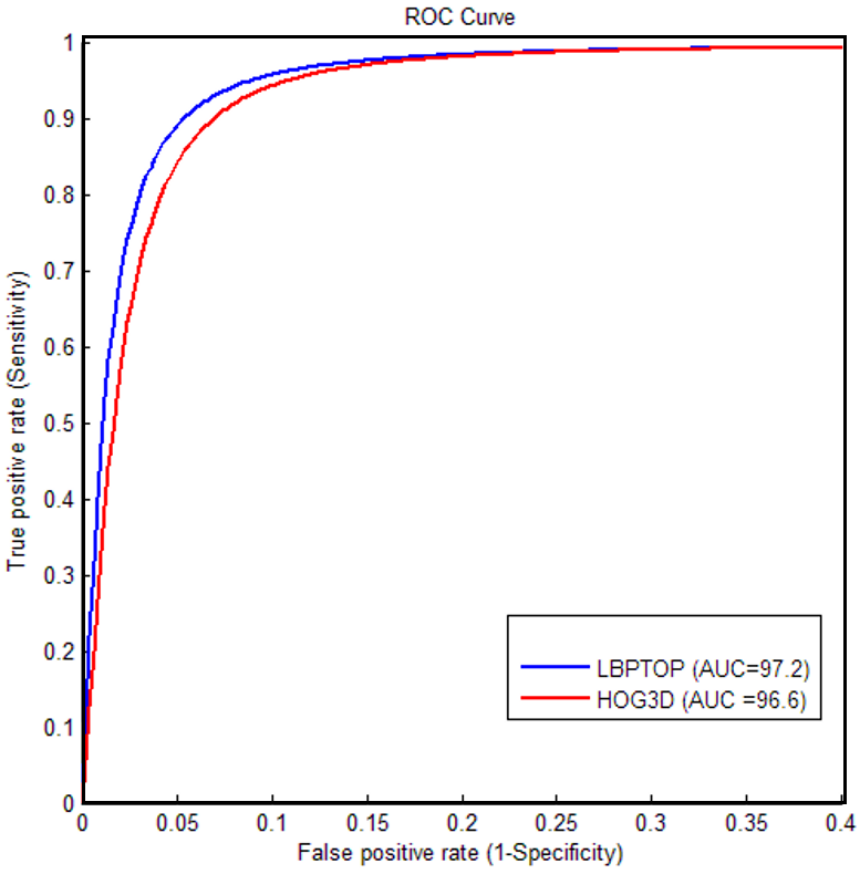


Figure 4.3: Cross validation ROC with HOG3D and LBP-TOP features

4.2.3 Performance evaluation

To evaluate the performance of the HOG3D and LBP-TOP features in CPR activity representation, we use the Receiver Operating Characteristic (ROC) curve. The ROC curve is a plot of true positive rate (TPR) vs false positive rate (FPR), for different values of discrimination threshold of the binary classifier. We measure the accuracy of classification in terms of area under the curve (AUC). AUC will have a value between 0 and 1, and is equivalent to the probability that the classifier will rank a randomly chosen positive instance higher than a randomly chosen negative instance.

The purpose of this experiment is to illustrate that the binary SVM classifier is capable of discriminating between a CPR and a non-CPR action cycle using the HOG3D and LBP-TOP features. Figure 4.3 shows the ROC curve obtained from a 10-fold cross

validation. With HOG3D we were able to achieve 96.6% cross validation accuracy with 5% false alarms and with LBP-TOP, we obtained an accuracy of 97.2% with 7% false alarms. It can be seen that both features can achieve high probability of detection with low probability of false alarms.

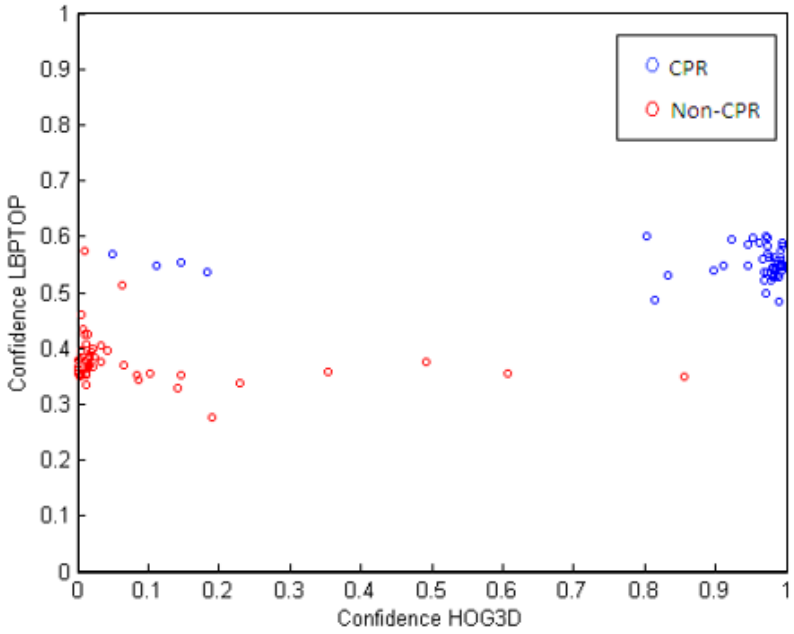


Figure 4.4: Confidence of HOG3D vs LBP-TOP with SVM classifier

The scatter plot of confidence of HOH3D vs confidence of LBP-TOP, for linear SVM classifier, is shown in figure 4.4. We see that some CPR instances are classified with high confidence using HOG3D features but not so with LBP-TOP features. Similarly, some non-CPR sequences are classified as CPR by HOG3D, but correctly identified by LBP-TOP features. Careful investigation of the detections from both features revealed that some CPR volume cycles that are missed by HOG3D SVM model can be detected by the LBP-TOP SVM model and vice versa, as illustrated in figures 4.5. Thus features from different modalities can provide complementary information. We utilize this characteristic to apply decision level fusion techniques, to improve the performance, as will be explained later in this chapter.

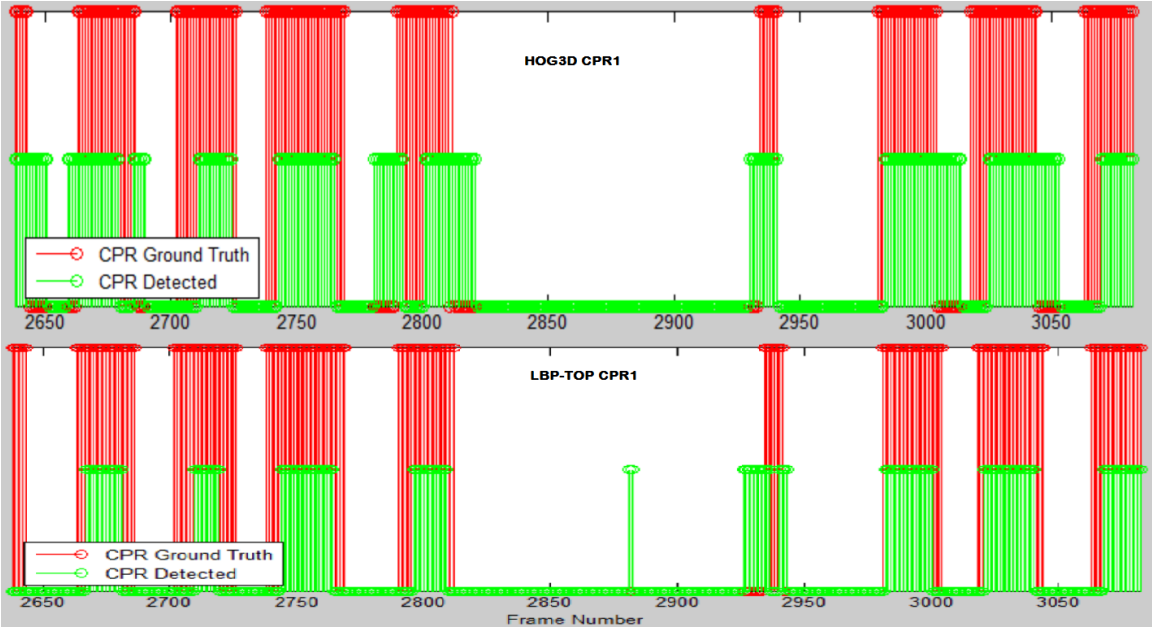


Figure 4.5: CPR1 retrieved scenes: HOG3D vs LBP-TOP

4.3 Analysis of the proposed system using unsegmented video streams

In this section, we explain how our framework can be used in a more realistic setting for CPR scene retrieval. In the previous experiment, we created the training data by manually extracting the bounding boxes around the CPR activity region. Manual video segmentation and scene selection is not practical and need to be automated. In the following experiments, we evaluate the performance of our models in detecting and retrieving the CPR scenes from unsegmented video streams.

As illustrated in figure 3.1, the first step of CPR scene retrieval consists of identifying the dummy's chest region. We process each video few frames at a time, starting from the first frame, using temporally overlapping windows, and the first frame of each volume is subjected to chest detection using the method explained in section 3.1.2.1. More specifically, we scan the entire test image using a 150×150 window with an overlap of 20 pixels, and a likelihood score is computed for every selected region. The area with the maximum likelihood score is identified as the chest and the region around it (200×300 pixels) is selected as the region of interest which is tested for the presence of CPR activity. First, we

conduct out experiments using the HOG3D features.

The identified region of interest is assumed to remain the same for 3 frames, which is the temporal overlap we allow for the volume window. The chest region is temporally divided into overlapping volumes of 18 frames each. Each of the volumes are further divided into $w \times h \times l$ overlapping grids. In our experiments, we set $w = h = 55$ and $l = 18$. We provide a 50% overlap within the grids. The HOG3D features are computed for each of the grid blocks. Each of these blocks are divided into $2 \times 2 \times 2$ sub-blocks, which results in a feature vector of $2 \times 2 \times 2 \times 20 = 160$ dimensions. This is the test dataset, Ts^{HOG3D}.

The test data is then classified using the trained binary SVM classifier described in section 4.2.1. Each volume is classified into CPR or non-CPR based on a certain classification confidence. Since we used overlapping video volumes, the confidence in adjacent video volumes are averaged out in the overlapping area. As a result we obtain a confidence value at each pixel, which spans over the entire volume of 18 frames in the given sequence. Any given video volume is classified into CPR action volume based on the average confidence of the positively classified grid window volumes in the scene.

The CPR activity detections from two CPR scenes are presented in figure 4.6. Only the positively classified grid windows are color-coded. Red color shows high confidence and blue color shows low confidence. We can see that the CPR activity region is classified as positive with higher confidence than the neighboring regions. It is because, the edge orientation changes in those regions in the temporal dimension, is similar to that of CPR activity, even though the edge orientations are essentially different in the spatial dimension.

Figure 4.6: Sample detection results from two different CPR scenes with HOG3D features (illustrated on first frame)

In the second experiment, we analyze the effectiveness of LBP-TOP features for CPR activity detection. The framework for extracting the LBP-TOP features is the same as that of the HOG3D. After detecting the chest region, we extract overlapping sub video volumes of size $55 \times 55 \times 18$. We choose a neighborhood of 8 pixels for the XY, XT, and YT planes. The neighborhood radii we used is $x_{radius} = y_{radius} = 1$ and $t_{radius} = 2$ and construct feature vectors of 177 dimensions. This will be our second set of test data Ts^{LBPTOP}. These features are classified using the trained binary LBP-TOP SVM model. The average confidence in overlapping sub volumes using the LBP-TOP features on two separate CPR scenes are shown in figure 4.7.

In addition to the SVM classifier, we test our framework using KNN and ANN classifiers. We use the HOG3D and LBP-TOP features extracted from sections 4.2.1 and 4.2.2 to learn two KNN classifiers (which will be referred to as KNN-H and KNN-L for HOG3D and LBP-TOP respecively) with $K = 3$; and we also train two shallow neural network classifiers with 10 hidden neurons (which will be referred to as ANN-H and ANN-L for HOG3D and LBP-TOP respecively) . The test data Ts^{HOG3D} and Ts^{LBPTOP} are classified with the two KNN and ANN models. The classification parameters used for different classifiers are listed in table 4.3.

The above experiments are conducted for all videos from CPR1 to CPR6 and their

TABLE 4.3

Parameters used for Different Classifiers

Classifier	Parameters
SVM-H	C=2
SVM-L	C=8
KNN-H	K=3
KNN-L	K=3
ANN-H	No. of hidden neurons = 10
ANN-H	No. of hidden neurons = 10

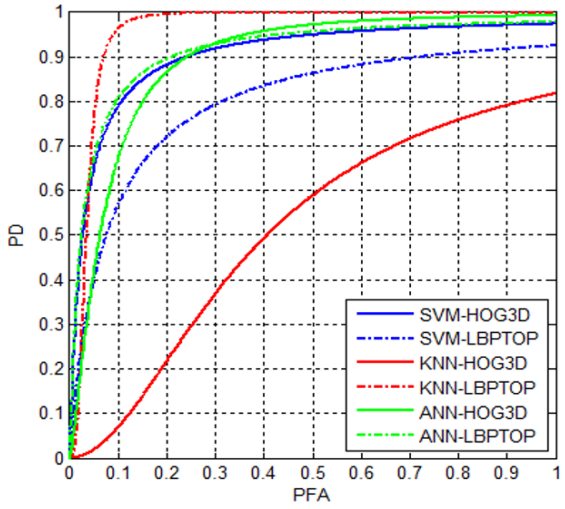
results are presented in the following subsection. The hand posture while performing CPR on an infant mannequin (in CPR7) is significantly different from the other videos where CPR is performed on child mannequin. Hence, we develop a different model for CPR7. Two - thirds of the CPR scenes are used for training and the testing is performed on the whole video.

Figure 4.7: Sample detection results from two different CPR scenes with LBP-TOP features (illustrated on first frame)

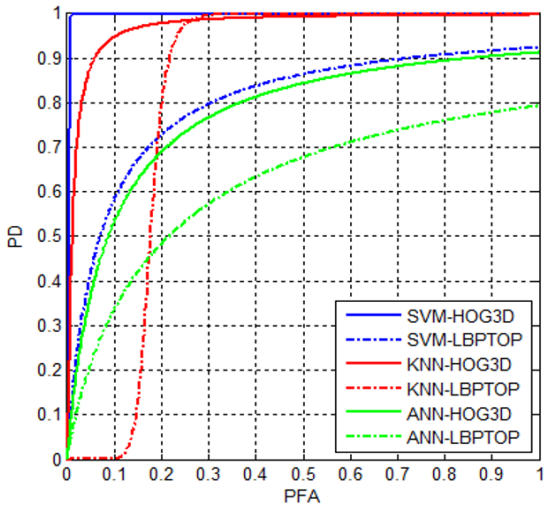
4.3.1 Performance Evaluation

To evaluate the performance of the proposed method to identify and retrieve CPR scenes, we use the receiver operating characteristic (ROC) curve. The ROC curves obtained using HOG3D and LBP-TOP features with SVM, KNN and ANN models are shown in figure 4.8.

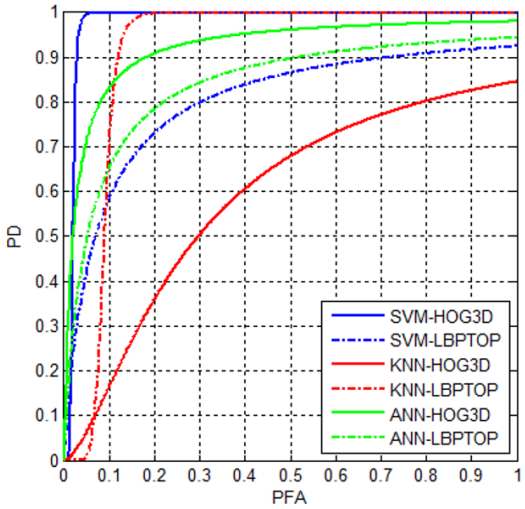
Video CPR8 has no CPR scenes, and the framework correctly classifies all scenes as



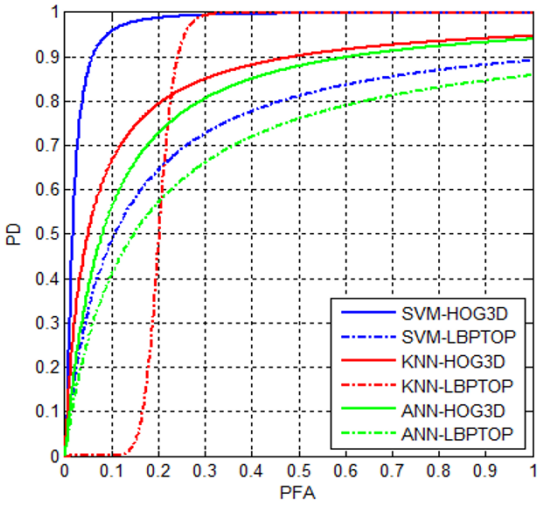
(a) ROC CPR1



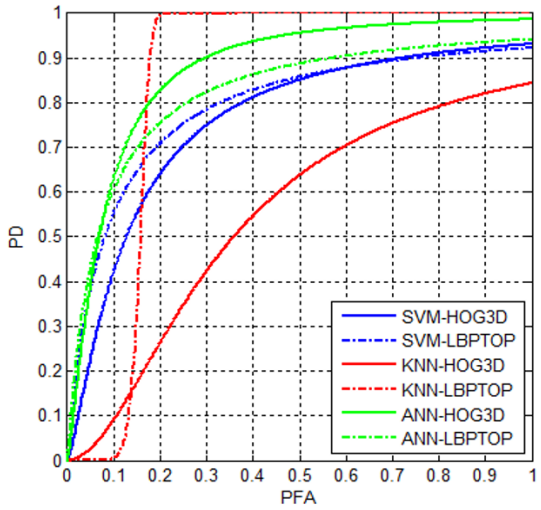
(b) ROC CPR2



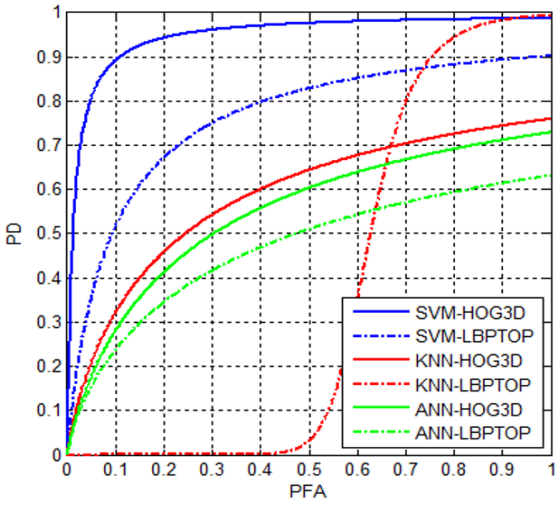
(c) ROC CPR3



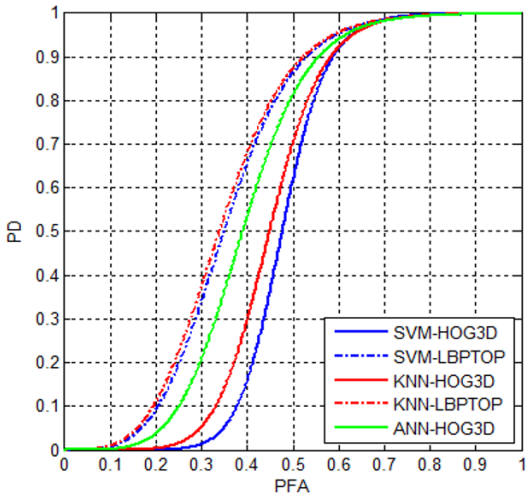
(d) ROC CPR4



(e) ROC CPR5



(f) ROC CPR6



(g) ROC CPR7

Figure 4.8: ROC generated from HOG3D and LBP-TOP features

non-CPR scenes. Due to the absence of true positives, we cannot report the result using ROC curve.

We see that both the features perform comparably with SVM classifier. The difference in performance is most for CPR3 and CPR6. In CPR3, the actors are back-facing the camera, so CPR actions are mostly occluded and hence HOG3D will not detect the structure of the hands performing CPR. Even if the hands are not visible, the back side of the actor moves in up-down cycle, while performing CPR. LBP-TOP captures most of this back-side movement as the CPR activity. On further inspection we also find that HOG3D is more sensitive to the rate of CPR compressions. It fails to capture very fast or very slow CPR, whereas LBP-TOP is not as sensitive to the rate of compressions. KNN classifier performs reasonably good with both the features, while ANN does not perform as good. In CPR7, the CPR action is performed using two or three fingers in the center of the chest and gentle compressions are given at the rate of $100 - 120/minute$. In most of the scenes, this action does not involve significant hand movement, and our framework does not recognize it as the CPR activity. The video CPR8 apparently does not include any CPR scenes and the system correctly does not retrieve any scene.

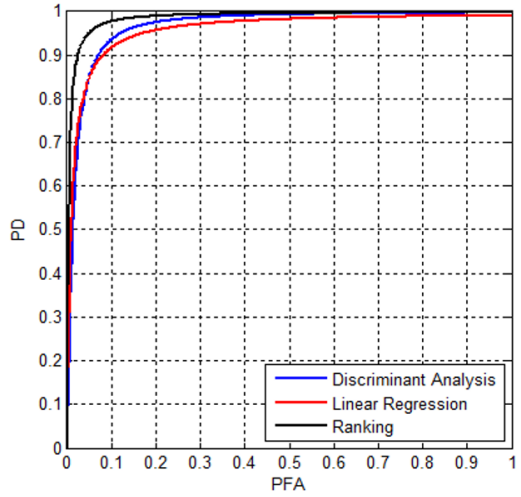
The current system inherently does not handle occlusion. However, since we use

overlapping video volumes, occlusions that happen in the middle of a CPR activity are ignored by the system. This means, if few volumes are classified as non-CPR volumes within a series of CPR volumes, we forcefully alter the class of intermediate volumes into CPR volumes. Since the objective of this work is to retrieve the scenes that specifically contain the CPR activity and not a frame by frame classification, the above work around is adopted to improve the system's efficiency in handling occlusions. We notice that sometimes actors perform CPR without noticeable movement of the hands. Such volumes are mostly classified with very low confidence, and are not retrieved by the system. Also, sometimes actors examine heartbeat and other vital pulses in the chest region, with their hands. These action continued for a fair duration, are in turn classified as CPR action and they mostly cause the false detections.

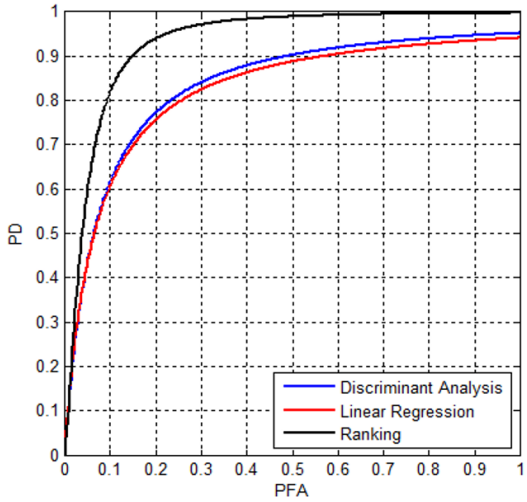
4.3.2 Decision-level fusion

In the next set of experiments, we perform decision level fusion of all the classifier outputs and analyze the behavior of different fusion techniques.The ROC curve obtained after decision fusion with three different algorithms namely, discriminant analysis, linear regression and ranking with worst performer removed, are shown in figure 4.9

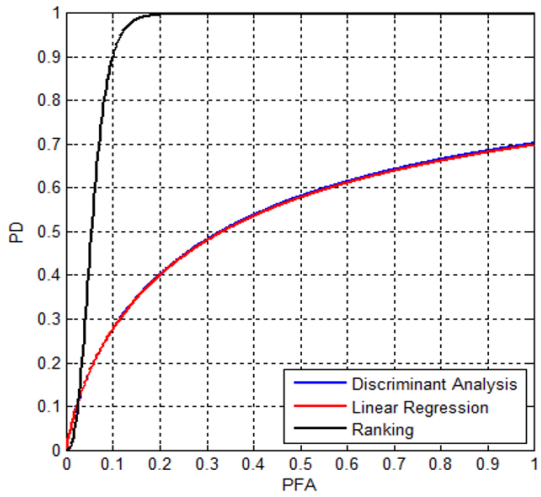
The AUC values of 6 videos, CPR1 to CPR6, using HOG3D and LBP-TOP methods with different classifiers are reported in table 4.4. The results of decision fusion using LBP-TOP and HOG3D features with SVM, KNN and ANN classifiers are presented in table 4.6. Of the three decision fusion techniques logistic regression (LR), discriminant analysis (DA) and ranking, we find that ranking with the worst performer (ANN-LBPTOP) removed produced the best results.



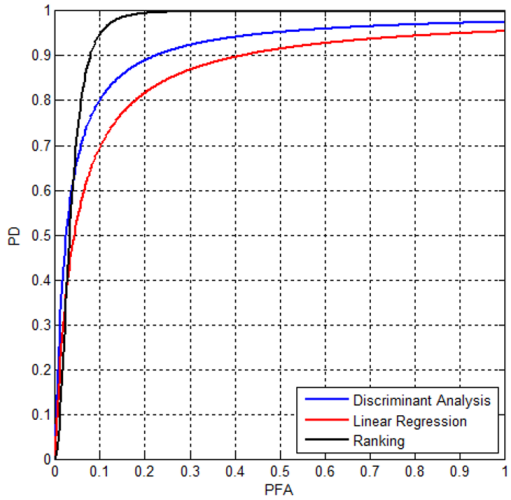
(a) ROC CPR1



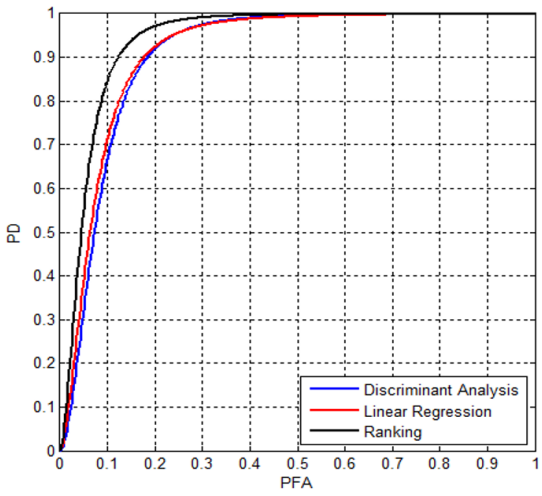
(b) ROC CPR2



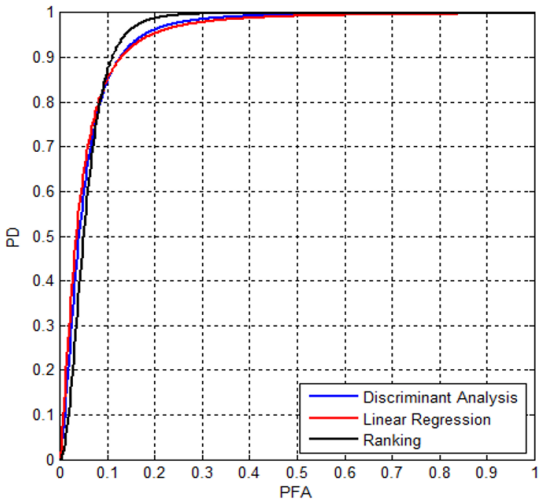
(c) ROC CPR3



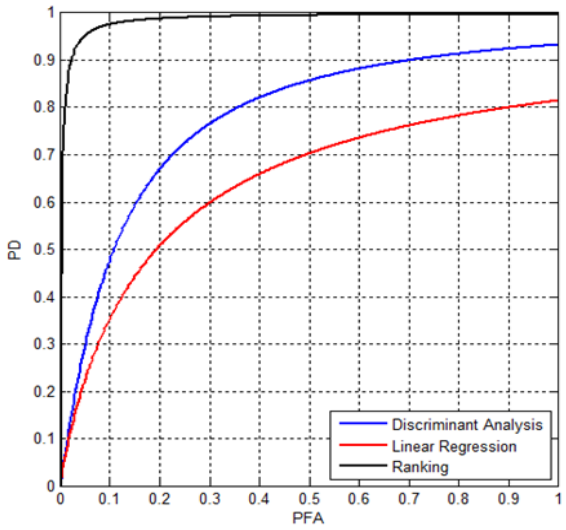
(d) ROC CPR4



(e) ROC CPR5



(f) ROC CPR6



(g) ROC CPR7

Figure 4.9: ROC after decision level fusion

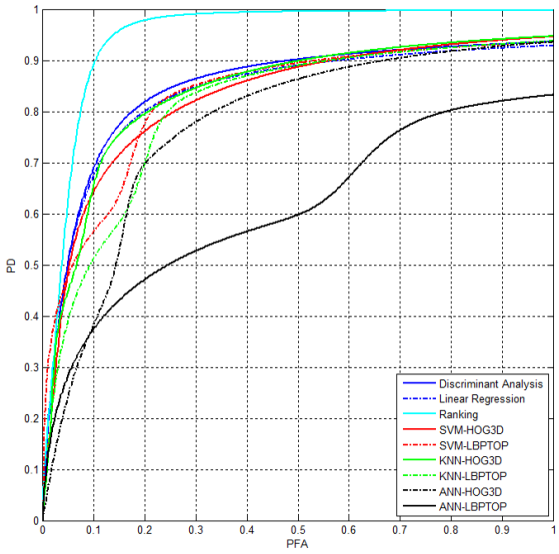


Figure 4.10: Mean ROC for 6 videos for all classifiers

The overall classification accuracy obtained was 95.25 %. The comparison of mean ROC obtained for 6 videos, for the different classifiers is shown in figure 4.10.

The performance of our system is better than the existing system [7], which has an average AUC of 70.3, for four videos. The accuracy of our system is greatly influenced by the presence of false positives. This is mainly because those regions have either their

TABLE 4.4

AUC for individual classifiers before decision fusion

Before Decision Fusion						
Video	SVM-H	SVM-L	KNN-H	KNN-L	ANN-H	ANN-L
CPR1	90.3	99.6	97.9	97.1	75.8	94.6
CPR2	79.5	79.8	79.9	74.2	78.8	76.0
CPR3	50.5	96.9	58.7	84.3	54.6	58.5
CPR4	95.9	82.0	90.8	79.7	84.4	36.0
CPR5	89.2	77.4	92.1	80.5	87.3	54.7
CPR6	91.2	61.4	83.7	68.9	81.9	46.5
Mean	82.8	82.9	83.9	80.8	77.1	61.0

TABLE 4.5

AUC for individual classifiers before decision fusion - CPR7

Before Decision Fusion						
Video	SVM-H	SVM-L	KNN-H	KNN-L	ANN-H	ANN-L
CPR7	52.0	64.0	54.6	61.1	64.8	60.0

TABLE 4.6

AUC after decision fusion

After Decision Fusion			
Video	DA	LR	Ranking
CPR1	96.6	95.8	97.8
CPR2	83.3	82.0	95.3
CPR3	52.8	52.6	94.1
CPR4	90.8	85.9	95.9
CPR5	90.6	91.3	94.3
CPR6	93.9	94.0	94.0
Mean	84.7	83.6	95.3

TABLE 4.7

AUC after decision fusion - CPR7

Video	DA	LR	Ranking
CPR7	77.3	63.5	98.4

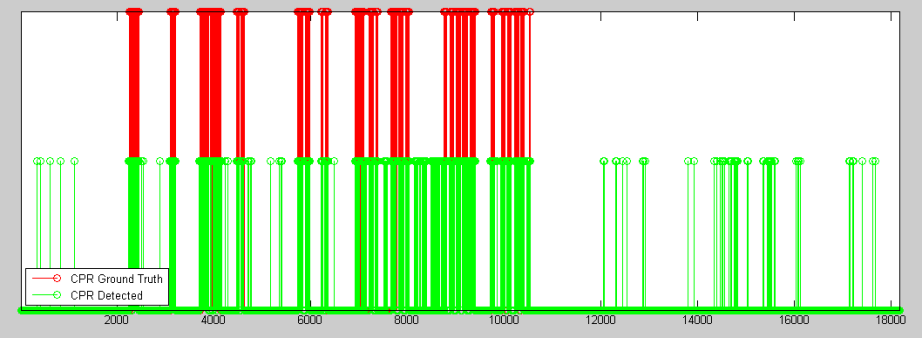


Figure 4.11: Stem plot showing ground truth and CPR scenes retrieved by the system for CPR1

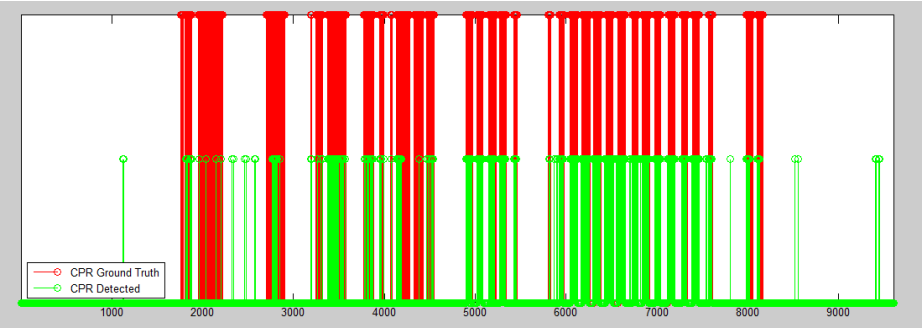


Figure 4.12: Stem plot showing ground truth and CPR scenes retrieved by the system for CPR2

shape structure or texture pattern, similar to that of the CPR volume. For example, the texture of the clothes worn by people have similar edge orientation histograms as that of hands performing CPR. During the CPR activity, the body of the actor, the body of the mannequin etc. move in a similar rhythmic fashion. The LBP-TOP descriptor cannot discriminate between the hand and mannequin, as the texture on the hand and mannequin are not very different from each other.

The stem plots showing the real CPR scenes and the retrieved CPR scenes by our system for each video are shown in figures 4.11 through 4.17. The ground truth information of CPR scenes is shown in red color and the retrieved CPR frames are shown in green. The magnitude of retrieved CPR frames in Y-axis is purposely reduced so as to illustrate the overlap of the retrieved frames and the ground truth.

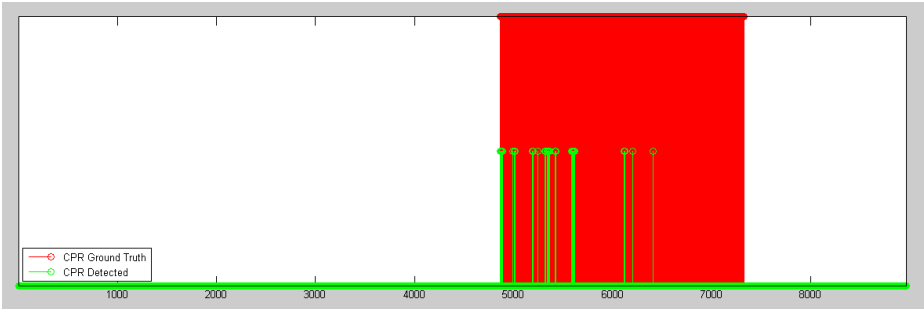


Figure 4.13: Stem plot showing ground truth and CPR scenes retrieved by the system for CPR3

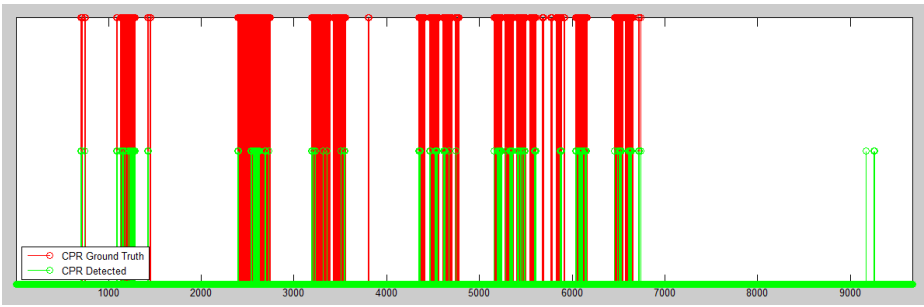


Figure 4.14: Stem plot showing ground truth and CPR scenes retrieved by the system for CPR4

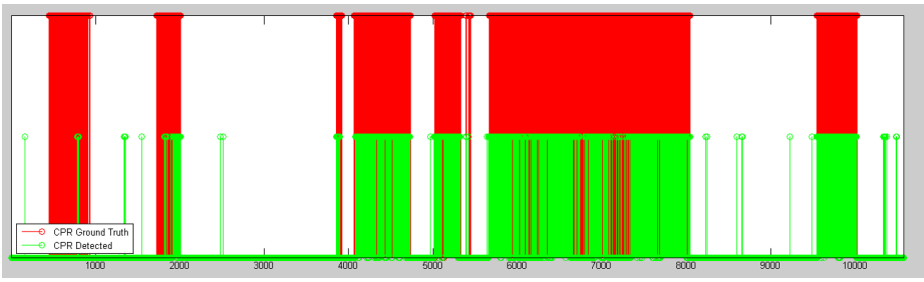


Figure 4.15: Stem plot showing ground truth and CPR scenes retrieved by the system for CPR5

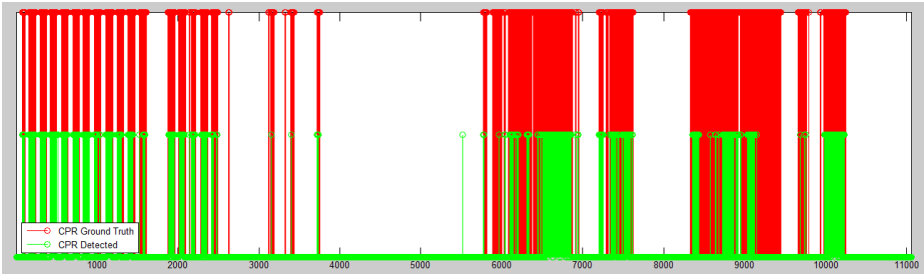


Figure 4.16: Stem plot showing ground truth and CPR scenes retrieved by the system for CPR6

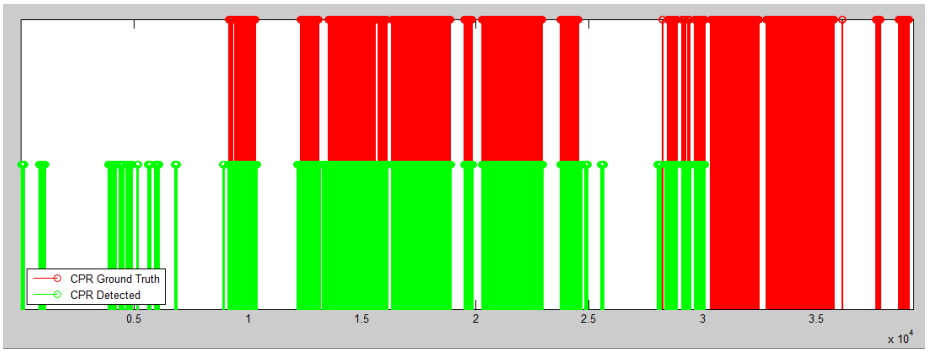


Figure 4.17: Stem plot showing ground truth and CPR scenes retrieved by the system for CPR7

4.4 Breathing bag activity detection

In this section, we explain our experiments for detecting the breathing bag activity. During this activity, the bag is squeezed, and the area of the detected bag reduces. Alternatively, the area of the bag increases when it is inflated. Our system detects this change in area and identify the presence of breathing bag activity in the corresponding scenes. Our approach analyzes every third frame of the video to identify the presence of breathing bag. We have noticed that this temporal sampling improves the speed without losing accuracy. First, we apply the breathing bag detection algorithm detailed in section 3.3.1.

The training phase consists of manually identifying few scenes that contain the breathing bag activity. The change in area of the breathing bag before and after the activity is computed. The area of a completely inflated breathing bag is approximately 1800 pixels and that of a deflated breathing bag is less than 800 pixels. Thus, we created a rule where, if the area of the detected breathing bag decreases by 400 pixels within 18 frames (numbers obtained empirically), it is considered as a deflation or a breathing bag activity. The frames where deflation happens are detected and the breathing bag activity scene is extracted as a video volume around this transition.

The duration of the breathing bag activity is not fixed. We can use the reduction of bag area to detect the squeezing action, but we cannot determine the duration of the activity. We can only detect the transition and extract a video volume around it to represent the breathing bag activity scene. We report the stem graph that shows the ground truth

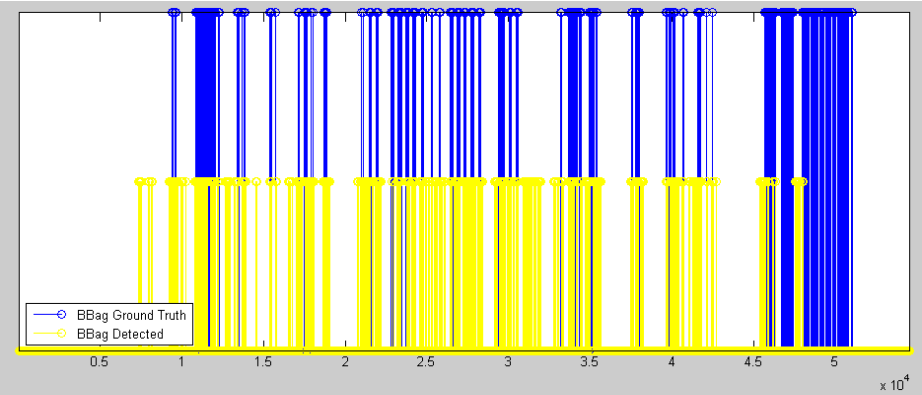


Figure 4.18: Stem plot showing ground truth and breathing bag scenes retrieved by the system for CPR1

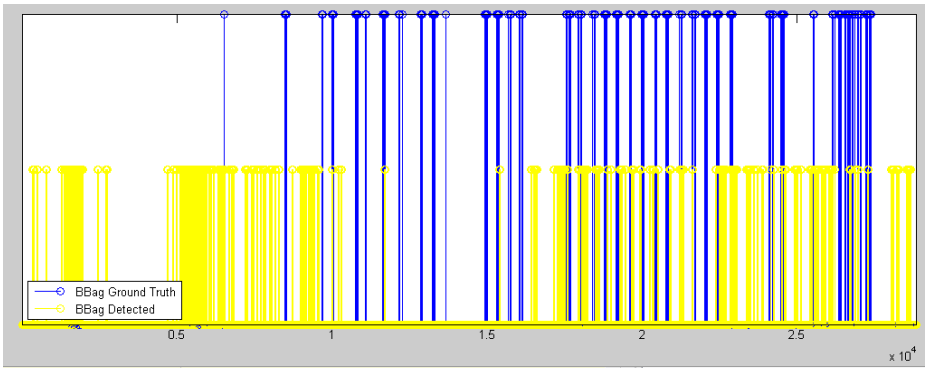


Figure 4.19: Stem plot showing ground truth and breathing bag scenes retrieved by the system for CPR2

breathing bag activity scenes and the breathing bag scenes retrieved by our system. These plots are shown in figures 4.18 through 4.23.

The main challenge with this approach is that, the breathing bag could be occluded by people moving in the scene, or by other objects in the scene. Sudden occlusion results in sudden reduction in the detected area, which is captured as the breathing bag activity, by the system.

The color of breathing bag used in CPR7 is different from that of the other videos. We developed a second color filter for CPR7 using the same method. The detection results are shown in figure 4.24.

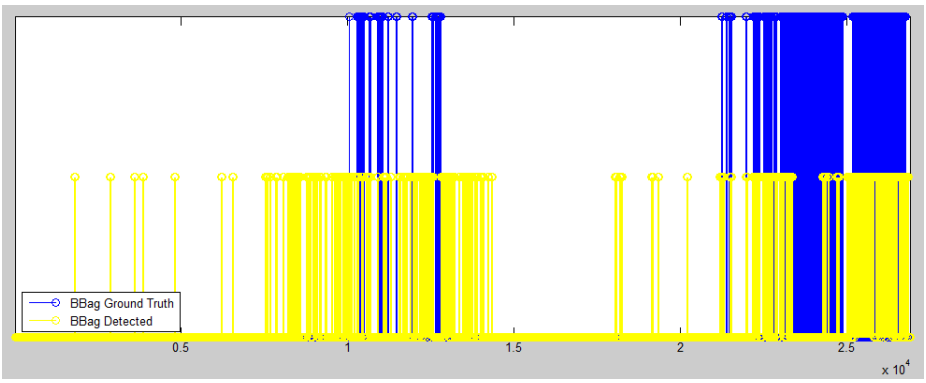


Figure 4.20: Stem plot showing ground truth and breathing bag scenes retrieved by the system for CPR3

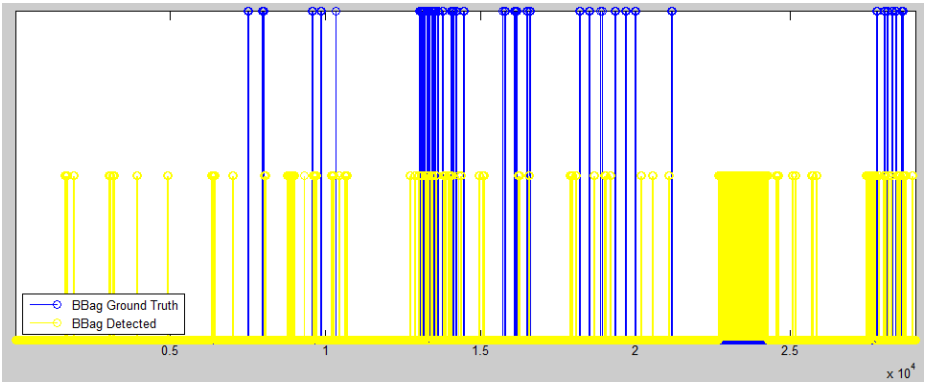


Figure 4.21: Stem plot showing ground truth and breathing bag scenes retrieved by the system for CPR4

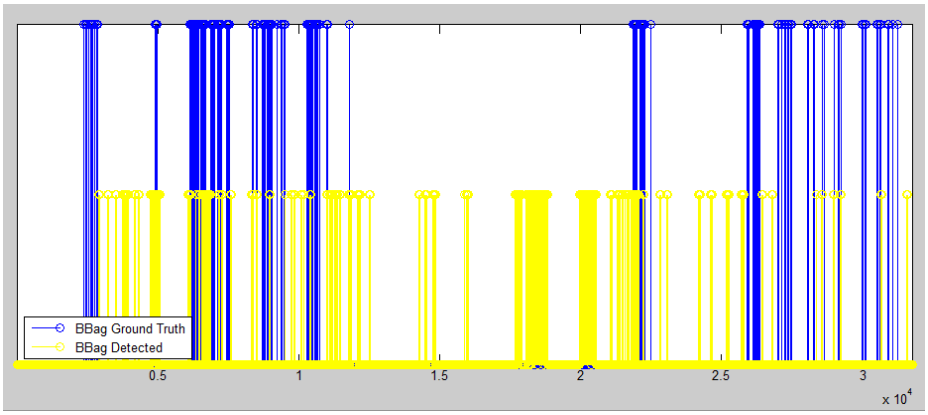


Figure 4.22: Stem plot showing ground truth and breathing bag scenes retrieved by the system for CPR5

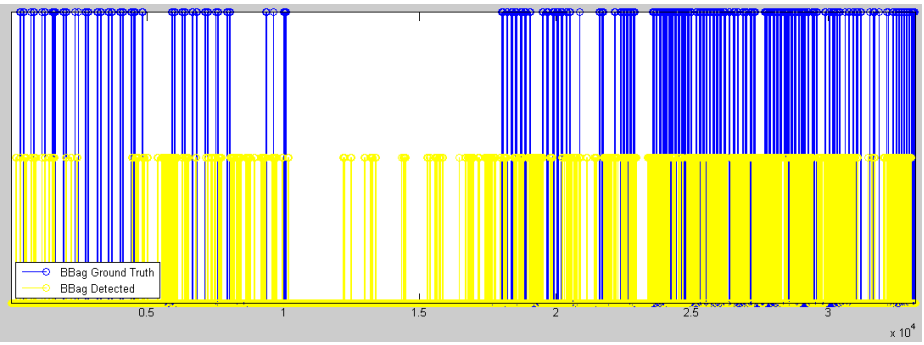


Figure 4.23: Stem plot showing ground truth and breathing bag scenes retrieved by the system for CPR6

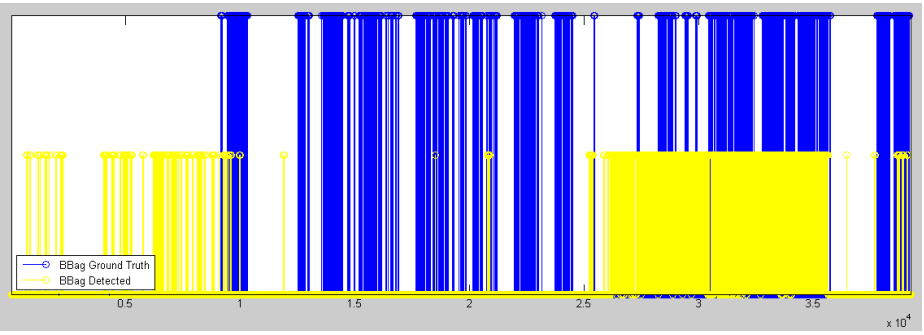


Figure 4.24: Stem plot showing ground truth and breathing bag scenes retrieved by the system for CPR7

4.5 A Graphical User Interface for CPR Scene Retrieval

To facilitate user interaction, we developed a Graphical User Interface (GUI) for our CPR scene retrieval system. A block diagram illustrating the functionalities of the GUI is shown in figure 4.25.

The GUI has several desirable properties which will allow the user to select and analyze the simulation video more effectively. The different functionalities of the GUI are illustrated in figure 4.26. The user will be able to select the video sequence using the "Open" command button (refer to steps 1 and 2 in figure 4.26) and watch a preview of the selected video in a windows media player. The general description and statistics of the video are displayed (refer to step 3 in figure 4.26) and the user is given additional options (refer to steps 4,5 and 6 in figure 4.26).

Figure 4.27 shows a different view presented to the user on selecting the "CPR

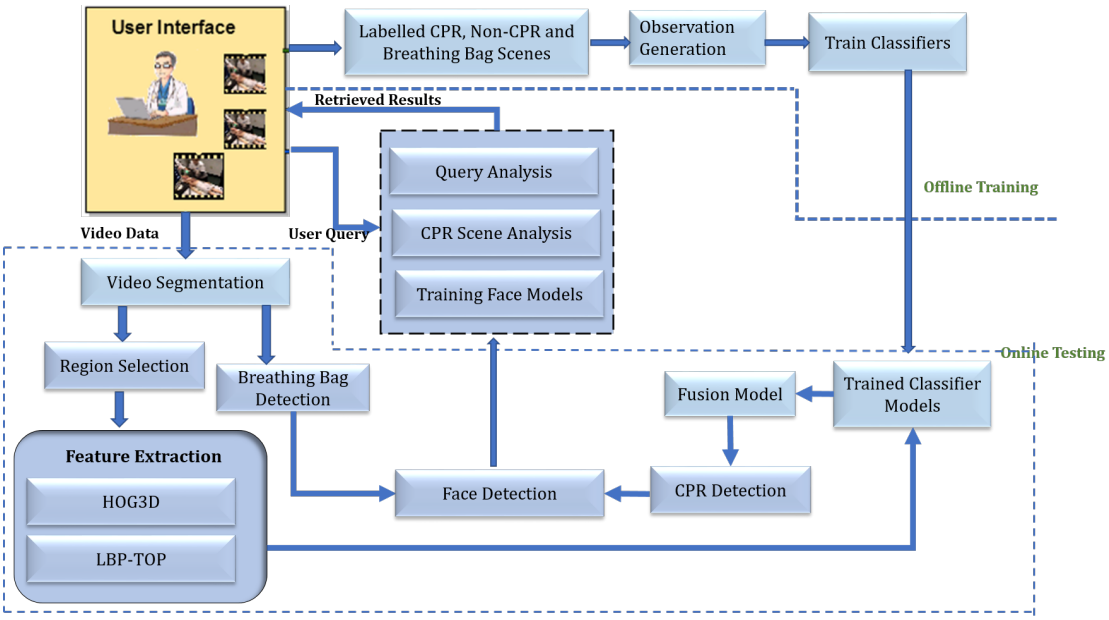


Figure 4.25: Block diagram of the proposed CPR scene retrieval prototype

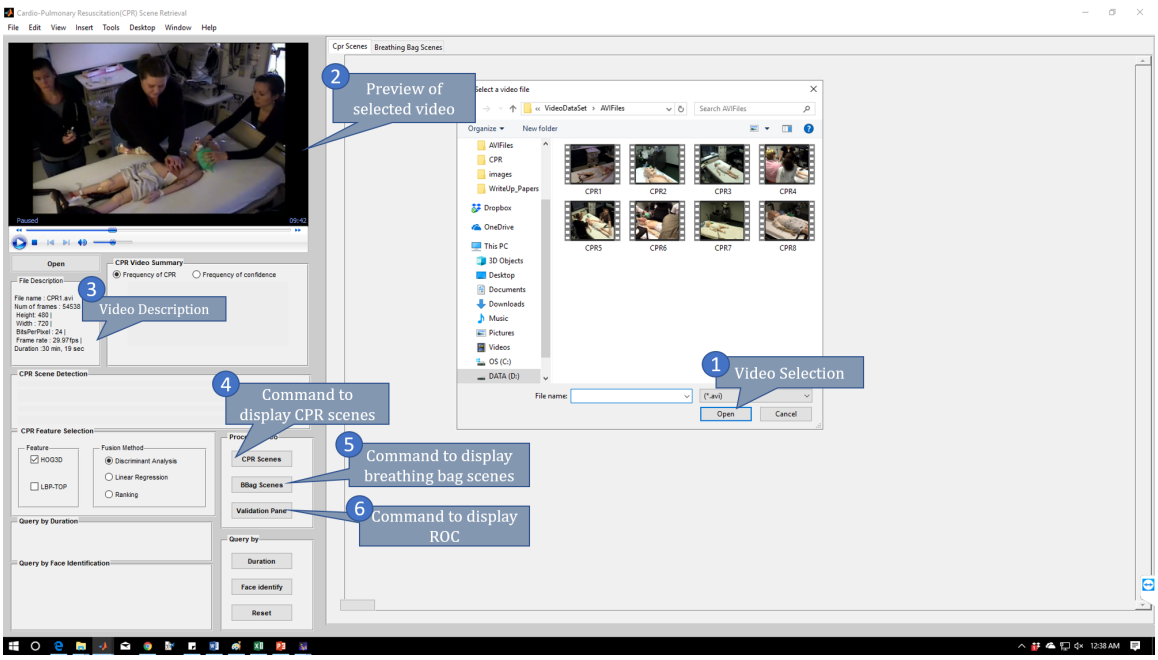


Figure 4.26: GUI of the proposed CPR scene retrieval prototype

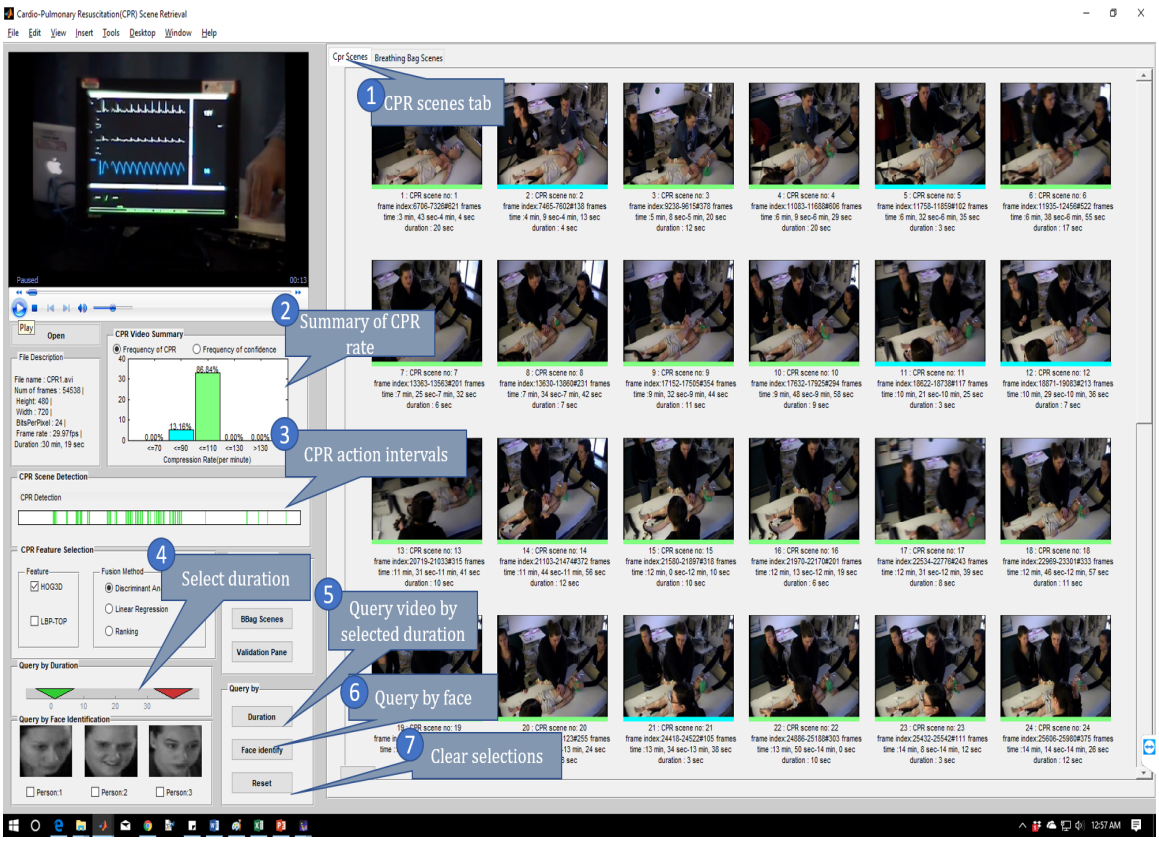


Figure 4.27: GUI for the analysis of the CPR scenes

Scenes" command button in step 4 in figure 4.26. In this view, the user has the option to play the full video on the top-left corner. The retrieved CPR scenes will be displayed on the "CPR Scenes" tab. The user can select and play each scene for further analysis. In addition to the CPR scenes, we provide a summary of the video statistics (refer to steps 2 and 3 in figure 4.27). The step 3 of figure 4.27 shows the temporal location of all CPR scenes in the video. In step 2 of figure 4.27, the user can visualize the histogram of the frequencies of all CPR scenes or histogram of confidence value assigned to all CPR scenes. For computing the rate of CPR compression we compute the motion vectors of all observations within the sequence, identify their zero crossings and determine the frequencies. The CPR rate is quantized into 5 discrete intervals as shown in table 4.8. This allows the physician to visualize only the CPR scenes within a given compression rate.

Figure 4.28 shows the breathing bag panel which is displayed on selecting the "Breathing Bag" command in step 5 of figure 4.26. This panel displays the retrieved breathing

TABLE 4.8

CPR Rate Quantization

Rate x	Quantized Value
x < 80	Very Slow
80 <= x < 95	Slow
95 <= x < 110	Good
110 <= x <125	Fast
x > 125	Very Fast

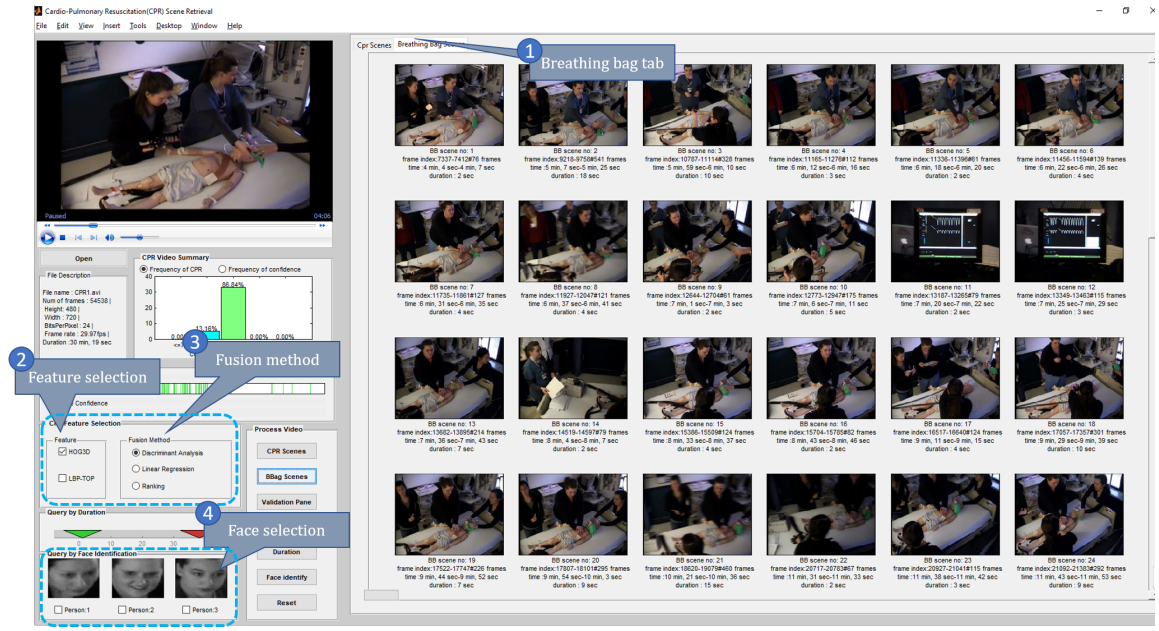


Figure 4.28: GUI for the analysis of breathing bag scenes

bag scenes. The user will be able to view the scene by clicking on the displayed thumbnail. Panel 4 of the GUI in figure 4.28 allows the user to view the CPR performed by a selected subject. For instance, if the user selects the 'Person 3', all the CPR performed by this subject will be filtered and displayed on the right side. The details of the query is shown in figure 4.29.

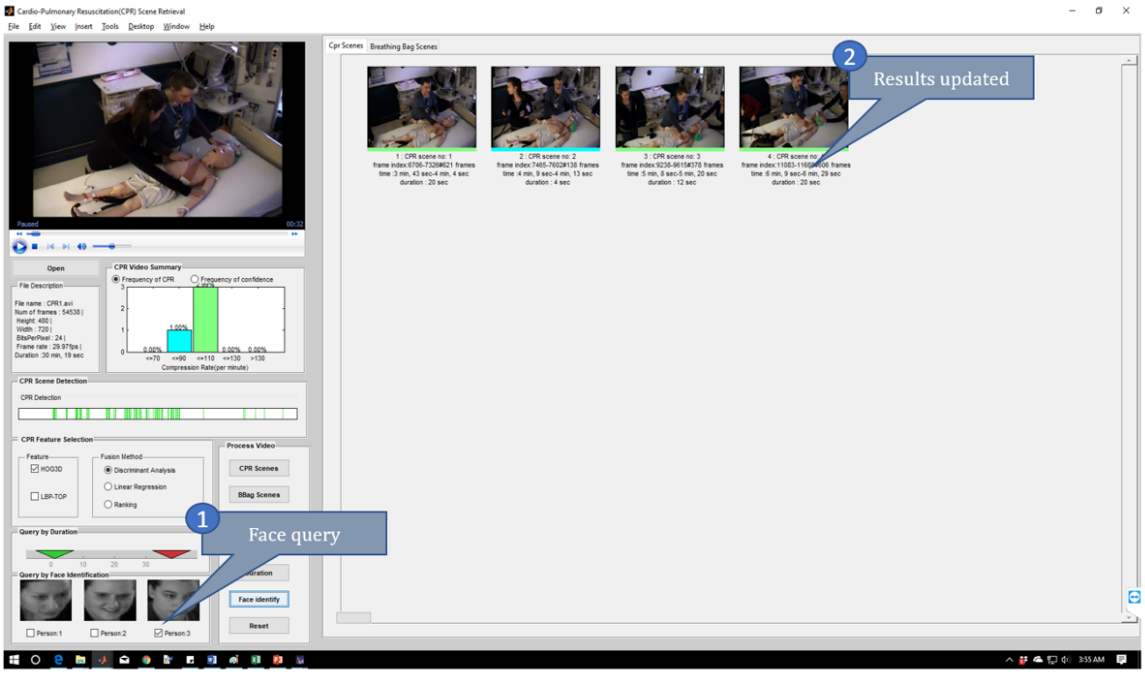


Figure 4.29: CPR performed by selected subject

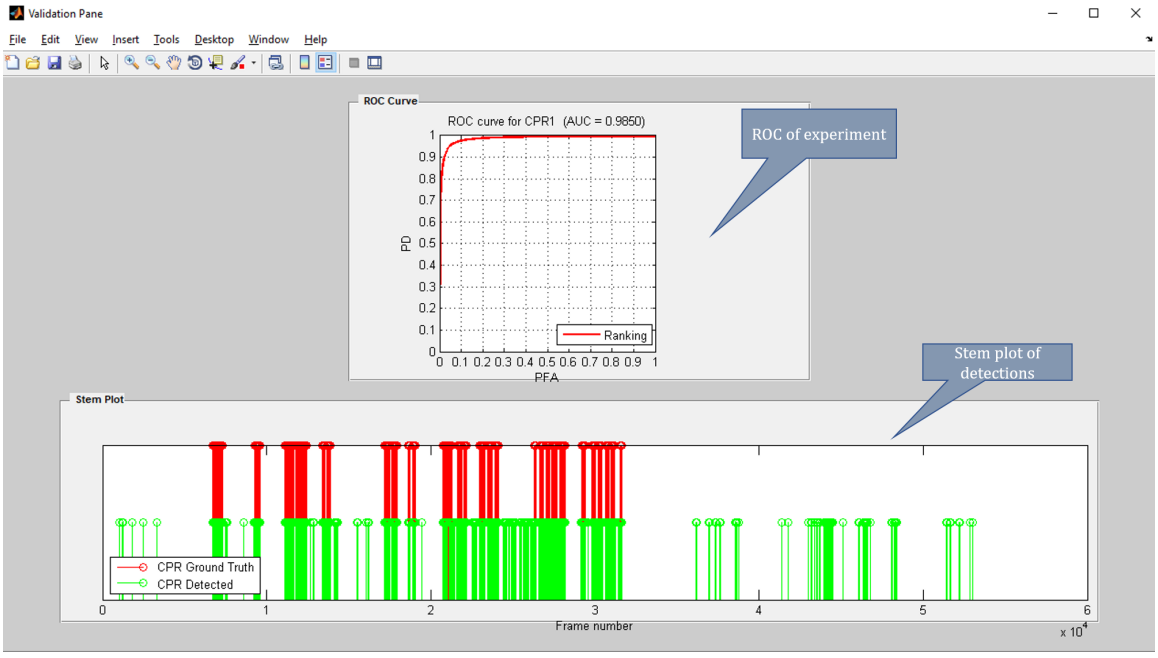


Figure 4.30: GUI of validation pane

Figure 4.30 shows the validation panel that is displayed by selecting "Validation" command from step 6 of figure 4.26. Here, the ROC and the stem plots of the selected

video and algorithm will be displayed.

CHAPTER 5

CONCLUSIONS AND POTENTIAL FUTURE WORK

5.1 Conclusions

We have proposed an approach for retrieving localized activities such as CPR activity scenes and breathing bag activity scenes from medical simulation videos. The proposed method of action-based scene retrieval, eliminates the need for video shot detection and frame segmentation phases, which are the typical preprocessing steps of most scene retrieval algorithms. We presented a framework that uses two spatio-temporal features namely, HOG3D and LBP-TOP for capturing the shape and dynamic texture of the CPR action cycle in three dimensions. For CPR activity detection, we first implemented a region of interest selection algorithm, which uses a skin pixel classifier to identify the chest region of the dummy. For validation we manually identify and label the CPR scenes and breathing bag scenes from the videos. Then, we built linear SVM models with HOG3D and LBP-TOP features, to differentiate between CPR and non-CPR activity. We also proposed a novel algorithm to detect breathing bag activity. Since breathing bags have a unique color we first create a color filter to detect the presence of breathing bag in the scene. Then, we detect the change in volume of the breathing bag caused by the squeezing, as the occurrence of the breathing bag activity.

The proposed approach was evaluated using eight 20 min video simulation sessions. These are simulated emergency room scenarios with multiple actors and multiple actions. For validation, we report and compare our results in the form of ROC curves and area under the ROC. We have shown that the HOG3D and LBP-TOP features achieve an average accuracy of 82.8% using a binary SVM classifier with linear kernel. Further, we evaluate our features with KNN and ANN classifiers to study the differences in classification accuracy

by different classification algorithms. KNN was able to achieve a mean accuracy of 83.9% with HOG3D and 80.8% with LBP-TOP features respectively. ANN classifier produce an average accuracy of 77.1% with HOG3D and61.0% with LBP-TOP features respectively.

Since the different features and classifiers can provide complementary information, we combined the outputs of these methods using three decision level fusion algorithms namely, linear regression, discriminant analysis and ranking. Our extensive experiments have indicated that the best fusion algorithm for CPR scene retrieval is the one based on combining the ranked values of different classifiers while ignoring the output of the worst classifier. We showed that fusion can achieve a scene retrieval accuracy of 94.4%.

We have developed a GUI for easy input and analysis if the videos. Using the proposed framework, the physician can have a quick access to specific scenes from a large collection of simulation videos. More importantly, our prototype GUI could be a very important tool in retrieving video scenes in response to high-level queries such as: "retrieve time-specic data about such critical events as elapsed time between failure of circulation and the initiation of CPR", a measure clearly associated with patient outcome.

Compared to a previously developed system for this application, our proposed method is simpler, more efficient, and more accurate. Our system is evaluated on a larger dataset of videos. Our approach has the advantage of avoiding common preprocessing and other intermediate steps as shot detection and image segmentation.

5.2 Potential Future Work

The proposed system could be improved further by:

1. Collecting and adopting the features and classifiers using a much larger data collector.
2. Enabling real-time processing of the videos using Graphical Processing Units (GPUs) or distributed computation (big data processing)

REFERENCES

[1] Limin Wang, Yu Qiao, and Xiaoou Tang, "Motionlets: Mid-level 3d parts for human motion recognition," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, June 2013, pp. 2674–2681.

[2] I. Laptev and T. Lindeberg, "Space-time interest points," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, Oct 2003, pp. 432–439 vol.1.

[3] Alexander Kläser, Marcin Marszalek, and Cordelia Schmid, "A spatio-temporal descriptor based on 3d-gradients," in *Proceedings of the British Machine Vision Conference 2008, Leeds, September 2008*, 2008, pp. 1–10.

[4] Yicong Tian, R. Sukthankar, and M. Shah, "Spatiotemporal deformable part models for action detection," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, June 2013, pp. 2642–2649.

[5] A.F. Bobick and J.W. Davis, "The recognition of human movement using temporal templates," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 23, no. 3, pp. 257–267, Mar 2001.

[6] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," in *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, Oct 2005, vol. 2, pp. 1395–1402 Vol. 2.

[7] Surangkana Rawungyot, *"Identification, indexing, and retrieval of cardio-pulmonary resuscitation (CPR) video scenes of simulated medical crisis."*, Ph.D. thesis, University of Louisville, 2014.

[8] YouTube, "Statistics," `https://www.youtube.com/yt/press/en-GB/statistics.html`, December 2014.

[9] P. Torrione and L. Collins, "Application of texture feature classification methods to landmine and clutter discrimination in off-road gpr data," in *Geoscience and Remote Sensing Symposium*, 2004, pp. 1621–1624 vol 1.

[10] B.V. Patel, A.V. Deorankar, and B.B. Meshram, "Content based video retrieval using entropy, edge detection, black and white color features," in *Computer Engineering and Technology (ICCET), 2010 2nd International Conference on*, April 2010, vol. 6, pp. V6–272–V6–276.

[11] Y.A. Aslandogan and C.T. Yu, "Techniques and systems for image and video retrieval," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 11, no. 1, pp. 56–63, Jan 1999.

[12] B. Huurnink, C.G.M. Snoek, M. de Rijke, and A.W.M. Smeulders, "Content-based analysis improves audiovisual archive retrieval," *Multimedia, IEEE Transactions on*, vol. 14, no. 4, pp. 1166–1178, Aug 2012.

[13] P. Ferguson, C. Gurrin, Hyowon Lee, S. Sav, A.F. Smeaton, N.E. O'Connor, Yoon-Hee Choi, and HeeSeon Park, "Enhancing the functionality of interactive tv with content-based multimedia analysis," in *Multimedia, 2009. ISM '09. 11th IEEE International Symposium on*, Dec 2009, pp. 495–500.

[14] M. Mori, H. Kimiyama, and M. Ogawara, "Search-based content analysis system on online collaborative platform for film production," in *Pattern Recognition (ICPR), 2014 22nd International Conference on*, Aug 2014, pp. 1091–1096.

[15] T.Q. Pham and L.J. van Vliet, "Separable bilateral filtering for fast video preprocessing," in *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*, July 2005, pp. 4 pp.–.

[16] C. Doutre and P. Nasiopoulos, "A colour correction preprocessing method for multiview video coding," in *Signal Processing Conference, 2008 16th European*, Aug 2008, pp. 1–5.

[17] Gu Jian-liu, "A processing method of atm monitoring video image under weak light environment," in *Knowledge Acquisition and Modeling (KAM), 2011 Fourth International Symposium on*, Oct 2011, pp. 270–273.

[18] W. Wattanarachothai and K. Patanukhom, "Key frame extraction for text based video retrieval using maximally stable extremal regions," in *Industrial Networks and Intelligent Systems (INISCom), 2015 1st International Conference on*, March 2015, pp. 29–37.

[19] T. Volkmer and A. Natsev, "Exploring automatic query refinement for text-based video retrieval," in *Multimedia and Expo, 2006 IEEE International Conference on*, July 2006, pp. 765–768.

[20] Liu Huayong, "Content-based tv sports video retrieval based on audio-visual features and text information," in *Web Intelligence, 2004. WI 2004. Proceedings. IEEE/WIC/ACM International Conference on*, Sept 2004, pp. 481–484.

[21] H. Miyamori, "Improving accuracy in behaviour identification for content-based retrieval by using audio and video information," in *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, 2002, vol. 2, pp. 826–830 vol.2.

[22] Xingquan Zhu, Xindong Wu, A.K. Elmagarmid, Zhe Feng, and Lide Wu, "Video data mining: semantic indexing and event detection from the association perspective," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 17, no. 5, pp. 665–677, May 2005.

[23] Zhao Yaqin, Zheng Jiaqiang, and Zhou Hongping, "News video clip retrieval based on topic caption text and audio information," in *Intelligent Systems, 2009. GCIS '09. WRI Global Congress on*, May 2009, vol. 4, pp. 477–481.

[24] Yongliang Xiao, "An effective video shot boundary detection method based on supervised learning," in *Advanced Computer Control (ICACC), 2010 2nd International Conference on*, March 2010, vol. 4, pp. 371–374.

[25] M. Rashid, S.A.R. Abu-Bakar, M. Mokji, and A. Abdu, "Human action concentric video retrieval system using features weight updating method as relevance feedback," in *Control System, Computing and Engineering (ICCSCE), 2012 IEEE International Conference on*, Nov 2012, pp. 366–370.

[26] Tianzhu Zhang, Changsheng Xu, Guangyu Zhu, Si Liu, and Hanqing Lu, "A generic framework for video annotation via semi-supervised learning," *Multimedia, IEEE Transactions on*, vol. 14, no. 4, pp. 1206–1219, Aug 2012.

[27] Yan Song, G.-J. Qi, Xian-Sheng Hua, Li-Rong Dai, and Ren-Hua Wang, "Video annotation by active learning and semi-supervised ensembling," in *Multimedia and Expo, 2006 IEEE International Conference on*, July 2006, pp. 933–936.

[28] Sangmin Oh and A. Hoogs, "Unsupervised learning of activities in video using scene context," in *Pattern Recognition (ICPR), 2010 20th International Conference on*, Aug 2010, pp. 3579–3582.

[29] M.M. Ben Ismail, O. Bchir, and A.Z. Emam, "Endoscopy video summarization based on unsupervised learning and feature discrimination," in *Visual Communications and Image Processing (VCIP), 2013*, Nov 2013, pp. 1–6.

[30] D. Geronimo and H. Kjellstrom, "Unsupervised surveillance video retrieval based on human action and appearance," in *Pattern Recognition (ICPR), 2014 22nd International Conference on*, Aug 2014, pp. 4630–4635.

[31] Hui-Yu Huang, Weir-Sheng Shih, and Wen-Hsing Hsu, "Movie classification using visual effect features," in *Signal Processing Systems, 2007 IEEE Workshop on*, Oct 2007, pp. 295–300.

[32] Min Xu, Jinqiao Wang, M.A. Hasan, Xiangjian He, Changsheng Xu, Hanqing Lu, and J.S. Jin, "Using context saliency for movie shot classification," in *Image Processing (ICIP), 2011 18th IEEE International Conference on*, Sept 2011, pp. 3653–3656.

[33] A. Ezzahout and R.O.H. Thami, "Conception and development of a video surveillance system for detecting, tracking and profile analysis of a person," in *ISKO-Maghreb, 2013 3rd International Symposium*, Nov 2013, pp. 1–5.

[34] M. Andersson, F. Gustafsson, L. St-Laurent, and D. Prevost, "Recognition of anomalous motion patterns in urban surveillance," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 7, no. 1, pp. 102–110, Feb 2013.

[35] newsdesk, "Security," `http://www.securitynewsdesk.com/`, June 2015.

[36] Yan Chen, Ling Zhang, Biyi Lin, Yong Xu, and Xiaobo Ren, "Fighting detection based on optical flow context histogram," in *Innovations in Bio-inspired Computing and Applications (IBICA), 2011 Second International Conference on*, Dec 2011, pp. 95–98.

[37] M. Elhamod and M.D. Levine, "Automated real-time detection of potentially suspicious behavior in public transport areas," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 14, no. 2, pp. 688–699, June 2013.

[38] Weihua Wang and Zhijing Liu, "A new approach for real-time detection of abandoned and stolen objects," in *Electrical and Control Engineering (ICECE), 2010 International Conference on*, June 2010, pp. 128–131.

[39] Jin Wang, Zhongqi Zhang, Bin Li, Sungyoung Lee, and R.S. Sherratt, "An enhanced fall detection system for elderly person monitoring using consumer home networks," *Consumer Electronics, IEEE Transactions on*, vol. 60, no. 1, pp. 23–29, February 2014.

[40] Zan Gao, M. Detyniecki, Ming yu Chen, Wen Wu, A.G. Hauptmann, and H.D. Wactlar, "Towards automated assistance for operating home medical devices," in *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*, Aug 2010, pp. 2141–2146.

[41] F. Hijaz, N. Afzal, T. Ahmad, and O. Hasan, "Survey of fall detection and daily activity monitoring techniques," in *Information and Emerging Technologies (ICIET), 2010 International Conference on*, June 2010, pp. 1–6.

[42] Baoxin Li and M.I. Sezan, "Event detection and summarization in sports video," in *Content-Based Access of Image and Video Libraries, 2001. (CBAIVL 2001). IEEE Workshop on*, 2001, pp. 132–138.

[43] Liu Huayong and Zhang Hui, "A content-based broadcasted sports video retrieval system using multiple modalities: Sportbr," in *Computer and Information Technology, 2005. CIT 2005. The Fifth International Conference on*, Sept 2005, pp. 652–656.

[44] Long Sha, P. Lucey, S. Sridharan, S. Morgan, and D. Pease, "Understanding and analyzing a large collection of archived swimming videos," in *Applications of Computer Vision (WACV), 2014 IEEE Winter Conference on*, March 2014, pp. 674–681.

[45] H. Bhaumik, S. Bhattacharyya, S. Dutta, and S. Chakraborty, "Towards redundancy reduction in storyboard representation for static video summarization," in *Advances in Computing, Communications and Informatics (ICACCI, 2014 International Conference on*, Sept 2014, pp. 344–350.

[46] Xiang Li, Xiao Lin, Yan Gao, Bin Sheng, and Lizhuang Ma, "Gpu-based motion blending for motion graphs," in *Computational and Information Sciences (ICCIS), 2011 International Conference on*, Oct 2011, pp. 955–959.

[47] Yuen May Chan, Koo Ah Choo, and P.C. Woods, "Youtube videos for learning principles of animation," in *Informatics and Creative Multimedia (ICICM), 2013 International Conference on*, Sept 2013, pp. 43–46.

[48] M. Sun, A.D. Jepson, and E. Fiume, "Video input driven animation (vida)," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, Oct 2003, pp. 96–103 vol.1.

[49] J.K. Aggarwal and Q. Cai, "Human motion analysis: a review," in *Nonrigid and Articulated Motion Workshop, 1997. Proceedings., IEEE*, Jun 1997, pp. 90–102.

[50] Qiang Liu, R.J. Sclabassi, and Mingui Sun, "A new change detection method and its application to epilepsy monitoring video," in *Bioengineering Conference, 2004. Proceedings of the IEEE 30th Annual Northeast*, April 2004, pp. 59–60.

[51] I. Chakraborty, A. Elgammal, and R.S. Burd, "Video based activity recognition in trauma resuscitation," in *Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on*, April 2013, pp. 1–8.

[52] B. Soran, Jenq-Neng Hwang, Su-In Lee, and L. Shapiro, "Tremor detection using motion filtering and svm," in *Pattern Recognition (ICPR), 2012 21st International Conference on*, Nov 2012, pp. 178–181.

[53] B. S. Abella, J. P. Alvarado, H. Myklebust, D. P. Edelson, A. Barry, N. OHearn, T. L. V. Hoek, and L. B. Becker, "Quality of cardiopulmonary resuscitation during in-hospital cardiac arrest," in *Journal of Americal Medial Association (JAMA)*, Jan 2005, pp. 305–310.

[54] Guoying Zhao and M. Pietikainen, "Dynamic texture recognition using local binary patterns with an application to facial expressions," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 6, pp. 915–928, June 2007.

[55] Florian Baumann, Jie Liao, Arne Ehlers, and Bodo Rosenhahn, "Motion binary patterns for action recognition," in *3rd International Conference on Pattern Recognition Applications and Methods*, Mar. 2014.

[56] L. Yeffet and L. Wolf, "Local trinary patterns for human action recognition," in *Computer Vision, 2009 IEEE 12th International Conference on*, Sept 2009, pp. 492–497.

[57] J.W. Davis, "Hierarchical motion history images for recognizing human motion," in *Detection and Recognition of Events in Video, 2001. Proceedings. IEEE Workshop on*, 2001, pp. 39–46.

[58] Ping Guo and Zhenjiang Miao, "Motion description with local binary pattern and motion history image: Application to human motion recognition," in *Haptic Audio visual Environments and Games, 2008. HAVE 2008. IEEE International Workshop on*, Oct 2008, pp. 171–174.

[59] A. Yilmaz and M. Shah, "Actions sketch: a novel action representation," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, June 2005, vol. 1, pp. 984–989 vol. 1.

[60] Lawrence Cayton, "Algorithms for manifold learning," *Univ. of California at San Diego Tech. Rep*, pp. 1–17, 2005.

[61] Mingyu Fan, Hong Qiao, Bo Zhang, and Xiaoqin Zhang, "Isometric multi-manifold learning for feature extraction," in *Data Mining (ICDM), 2012 IEEE 12th International Conference on*, Dec 2012, pp. 241–250.

[62] K.J. Cannons and R.P. Wildes, "The applicability of spatiotemporal oriented energy features to region tracking," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 36, no. 4, pp. 784–796, April 2014.

[63] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, June 2008, pp. 1–8.

[64] A. Utasi and L. Czuni, "Hmm-based unusual motion detection without tracking," in *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, Dec 2008, pp. 1–4.

[65] P. Saisan, G. Doretto, Ying Nian Wu, and S. Soatto, "Dynamic texture recognition," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, 2001, vol. 2, pp. II–58–II–63 vol.2.

[66] A.B. Chan and N. Vasconcelos, "Classifying video with kernel dynamic textures," in *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, June 2007, pp. 1–6.

[67] A.B. Chan and N. Vasconcelos, "Modeling, clustering, and segmenting video with mixtures of dynamic textures," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 30, no. 5, pp. 909–926, May 2008.

[68] I. Laptev and T. Lindeberg, "Space-time interest points," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, Oct 2003, pp. 432–439 vol.1.

[69] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features," in *Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005. 2nd Joint IEEE International Workshop on*, Oct 2005, pp. 65–72.

[70] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, June 2008, pp. 1–8.

[71] Xingxing Wang, LiMin Wang, and Yu Qiao, "A comparative study of encoding, pooling and normalization methods for action recognition," in *Proceedings of the 11th Asian Conference on Computer Vision - Volume Part III*, Berlin, Heidelberg, 2013, ACCV'12, pp. 572–585, Springer-Verlag.

[72] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, June 2005, vol. 1, pp. 886–893 vol. 1.

[73] R. Chaudhry, A. Ravichandran, G. Hager, and R. Vidal, "Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, June 2009, pp. 1932–1939.

[74] A. Satpathy, Xudong Jiang, and How-Lung Eng, "Human detection by quadratic classification on subspace of extended histogram of gradients," *Image Processing, IEEE Transactions on*, vol. 23, no. 1, pp. 287–297, Jan 2014.

[75] Haoyu Ren, Cher-Keng Heng, Wei Zheng, Luhong Liang, and Xilin Chen, "Fast object detection using boosted co-occurrence histograms of oriented gradients," in *Image Processing (ICIP), 2010 17th IEEE International Conference on*, Sept 2010, pp. 2705–2708.

[76] Paul Viola and Michael Jones, "Robust real-time object detection," in *International Journal of Computer Vision*, 2001.

[77] J. Fehr and H. Burkhardt, "3d rotation invariant local binary patterns," in *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, Dec 2008, pp. 1–4.

[78] M. Topi, O. Timo, P. Matti, and S. Maricor, "Robust texture classification by subsets of local binary patterns," in *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, 2000, vol. 3, pp. 935–938 vol.3.

[79] A. Ghahremani and A. Mousavinia, "Visual object tracking using kalman filter, mean shift algorithm and spatiotemporal oriented energy features," in *Computer and Knowledge Engineering (ICCKE), 2014 4th International eConference on*, Oct 2014, pp. 625–629.

[80] Liang Wang, Yizhou Wang, Tingting Jiang, and Wen Gao, "Instantly telling what happens in a video sequence using simple features," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, June 2011, pp. 3257–3264.

[81] A.F. Bobick and J.W. Davis, "The recognition of human movement using temporal templates," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 23, no. 3, pp. 257–267, Mar 2001.

[82] S. Sadanand and J.J. Corso, "Action bank: A high-level representation of activity in video," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, June 2012, pp. 1234–1241.

[83] A. Dargazany and M. Nicolescu, "Human body parts tracking using torso tracking: Applications to activity recognition," in *Information Technology: New Generations (ITNG), 2012 Ninth International Conference on*, April 2012, pp. 646–651.

[84] E-Jui Weng and Li-Chen Fu, "On-line human action recognition by combining joint tracking and key pose recognition," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, Oct 2012, pp. 4112–4117.

[85] M.B. Kaaniche and F. Bremond, "Tracking hog descriptors for gesture recognition," in *Advanced Video and Signal Based Surveillance, 2009. AVSS '09. Sixth IEEE International Conference on*, Sept 2009, pp. 140–145.

[86] A.J. Ma, P.C. Yuen, W.W.W. Zou, and Jian-Huang Lai, "Supervised spatio-temporal neighborhood topology learning for action recognition," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 23, no. 8, pp. 1447–1460, Aug 2013.

[87] Bi Song, N. Vaswani, and A.K. Roy-Chowdhury, "Closed-loop tracking and change detection in multi-activity sequences," in *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, June 2007, pp. 1–8.

[88] P. Matikainen, M. Hebert, and R. Sukthankar, "Trajectons: Action recognition through the motion analysis of tracked features," in *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, Sept 2009, pp. 514–521.

[89] U. Gargi, R. Kasturi, and S.H. Strayer, "Performance characterization of video-shot-change detection methods," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 10, no. 1, pp. 1–13, Feb 2000.

[90] Zhi-Cheng Zhao and An-Ni Cai, "Shot boundary detection algorithm in compressed domain based on adaboost and fuzzy theory," in *Advances in Natural Computation*, Licheng Jiao, Lipo Wang, Xinbo Gao, Jing Liu, and Feng Wu, Eds., vol. 4222 of *Lecture Notes in Computer Science*, pp. 617–626. Springer Berlin Heidelberg, 2006.

[91] Jianbo Shi and Jitendra Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.

[92] Jr. Forney, G.D., "The viterbi algorithm," *Proceedings of the IEEE*, vol. 61, no. 3, pp. 268–278, March 1973.

[93] Federico Semeraro, Antonio Frisoli, Claudio Loconsole, Filippo Bann, Gaetano Tammaro, Guglielmo Imbriaco, Luca Marchetti, and Erga L. Cerchiari, "Simulation and education," *Resuscitation*, vol. 84, no. 4, pp. 501–507, 2013.

[94] S. Schroder, N. Loftfield, B. Langmann, K. Frank, and E. Reithmeier, "Contactless operating table control based on 3d image processing," in *Engineering in Medicine and Biology Society (EMBC), 2014 36th Annual International Conference of the IEEE*, Aug 2014, pp. 388–392.

[95] D. Sarwinda and A. Bustamam, "Detection of alzheimer's disease using advanced local binary pattern from hippocampus and whole brain of mr images," in *International Joint Conference on Neural Networks (IJCNN), Vancouver, BC*, 2016, pp. 5051–5056.

[96] R. A. Binsaadoon and E. S. M. El-Alfy, "Flgbp: Improved method for gait representation and recognition," in *World Symposium on Computer Applications and Research (WSCAR), Cairo*, 2016, pp. 59–64.

[97] R. A. Khan, A. Meyer, H. Konik, and S. Bouakaz, "Pain detection through shape and appearance features," in *IEEE International Conference on Multimedia and Expo (ICME), San Jose, CA*, 2013, pp. 1–6.

[98] Ming-yu Chen and Alexander Hauptmann, "Mosift: Recognizing human actions in surveillance videos," 2009.

[99] A. Sami, N.B. Karayiannis, J.D.Jr. Frost, M.S. Wise, and E.M. Mizrahi, "Automated tracking of multiple body parts in video recordings of neonatal seizures," in *Biomedical Imaging: Nano to Macro, 2004. IEEE International Symposium on*, April 2004, pp. 312–315 Vol. 1.

[100] U. G. Mangai, S. Samanta, S.and Das, and Chowdhury P.R., "A survey of decision fusion and feature fusion strategies for pattern classification," in *IETE Technical Review*, 2010, vol. 27.

[101] Ludmila I. Kuncheva, James Bezdek, and Robert Duin, "Decision templates for multiple classifier fusion: An experimental comparison," in *Pattern Recognition*, 02 2001, vol. 34, pp. 299–314.

[102] K. Sirlantzis, S. Hoque, and M. C. Fairhurst, "Trainable multiple classifier schemes for handwritten character recognition," 06 2002.

[103] F. Huenupn, N. Yoma, C. Molina, and C. Garretn, "Confidence based multiple classifier fusion in speaker verification," in *Pattern Recognition Letters*, 05 2008, vol. 29, pp. 957–966.

[104] Y. Bi, T. Anderson, and S. McClean, "Combining rules for text categorization using dempster's rule of combination," in *Intelligent Data Engineering and Automated Learning*. 2004, pp. 457–463, Springer Berlin Heidelberg.

[105] T. K. Ho, J. J. Hull, and S. N. Srihari, "Decision in multiple classifier systems," in *IEEE Trans on Pattern Analysis and Machine Intelligence*, Jan 1994, vol. 16, pp. 66–75.

[106] G. J. McLachlan, "Discriminant analysis and statistical pattern recognition," 1992.

[107] S. Balakrishnama and A. Ganapathiraju, "Linear discriminant analysisa brief tutorial," `http://www.isip.msstate.edu/publications/reports/isip_internal/1998/linear_discrim_analysis/lda_theory.pdf`.

[108] Ronald A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals Eugen.*, vol. 7, pp. 179–188, 1936.

[109] Berthold K.P. Horn and Brian G. Schunck, "Determining optical flow," Tech. Rep., Cambridge, MA, USA, 1980.

[110] Y. Freund and R. E. Schapire, "A short introduction to boosting," 1999.

[111] Vitomir Štruc and Nikola Pavešić, "The complete gabor-fisher classifier for robust face recognition," *EURASIP Advances in Signal Processing*, vol. 2010, pp. 26, 2010.

[112] Vitomir Štruc and Nikola Pavešić, "Gabor-based kernel partial-least-squares discrimination features for face recognition," *Informatica (Vilnius)*, vol. 20, no. 1, pp. 115138, 2009.

[113] Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, New York, NY, USA, 1992, COLT '92, pp. 144–152, ACM.

[114] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, January 1967.

[115] W. L. J. Bussmann, J. B. J.and Martens, J. H. M. Tulen, F. C. Schasfoort, H. J. G. van den Berg-Emons, and H. J. Stam, "Measuring daily behavior using ambulatory accelerometry: The activity monitor," *Behavior Research Methods, Instruments, & Computers*, vol. 33, no. 3, pp. 349–356, Aug 2001.

[116] F. Foerster, M. Smeja, and J. Fahrenberg, "Detection of posture and motion by accelerometry: a validation study in ambulatory monitoring," *Computers in Human Behavior*, vol. 15, no. 5, pp. 571 – 583, 1999.

[117] R. Lippmann, "An introduction to computing with neural nets," *IEEE ASSP Magazine*, vol. 4, no. 2, pp. 4–22, Apr 1987.

[118] Xin Yao, "Evolving artificial neural networks," *Proceedings of the IEEE*, vol. 87, no. 9, pp. 1423–1447, Sep 1999.

[119] S. Oniga and J. Suto, "Activity recognition in adaptive assistive systems using artificial neural networks," vol. 22, 02 2016.

[120] Chih-Chung Chang and Chih-Jen Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011, Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[121] F. Baumann, A. Ehlers, B. Rosenhahn, and Jie Liao, "Computation strategies for volume local binary patterns applied to action recognition," in *Advanced Video and Signal Based Surveillance (AVSS), 2014 11th IEEE International Conference on*, Aug 2014, pp. 68–73.

CURRICULUM VITAE

NAME: Anju Panicker Madhusoodhanan Sathik

ADDRESS: Computer Engineering & Computer Science Department
Speed School of Engineering
University of Louisville
Louisville, KY 40292

EDUCATION:

Ph.D., Computer Science & Engineering
April 2018
University of Louisville, *Louisville, Kentucky*

M.S., Computer Science & Engineering
April 2012
University of Louisville, *Louisville, Kentucky*

B.Tech., Electronics and Communications Engineering
May 2006
University of Kerala, *Kerala, India*

PROFESSIONAL EXPERIENCE:

1. **Stubhub Inc, eBay**, Technical PhD Intern
   May 2017 August 2017, San Fransisco, CA.
2. **Genscape Inc**, Research & Development Intern
   June 2015 August 2017, Louisville, KY.

3. **Infosys Technologies Ltd.** , Software Engineer

   August 2006 July 2009, Trivandrum, India.

CONFERENCE PUBLICATIONS:

1. **M. S. A. Panicker**, H. Frigui and A. W. Calhoun, "Identification of cardio-pulmonary resuscitation (CPR) scenes in medical simulation videos using spatio-temporal gradient orientations," 2015 International Conference on Image Processing Theory, Tools and Applications (IPTA), Orleans, 2015.

2. **M. S. A. Panicker**, F. Tafazzoli, H. Frigui. CPR Activity Detection and Scene Retrieval from Medical Simulation Videos using Multiple Instance Learning. Poster at CVPR workshop, 2016.

3. Desoky, A.,**Madhusoodhanan, A.P.** Bitwise Hill Crypto System, Signal Processing and Information Technology (ISSPIT), Dec 2011.

HONORS AND AWARDS:

1. Grace Hopper Scholarship, 2016

2. First Prize, Student Research competition, Engineering Exposition, U of L, 2016

3. Grosscurth Fellowship for PhD, Uof L, 2012 to 2014

4. CECS Master of Science Award for Highest Cumulative Standing, 2011

5. Spot Award for Best Performance, Infosys Technologies, 2007